intel

Flash Memory

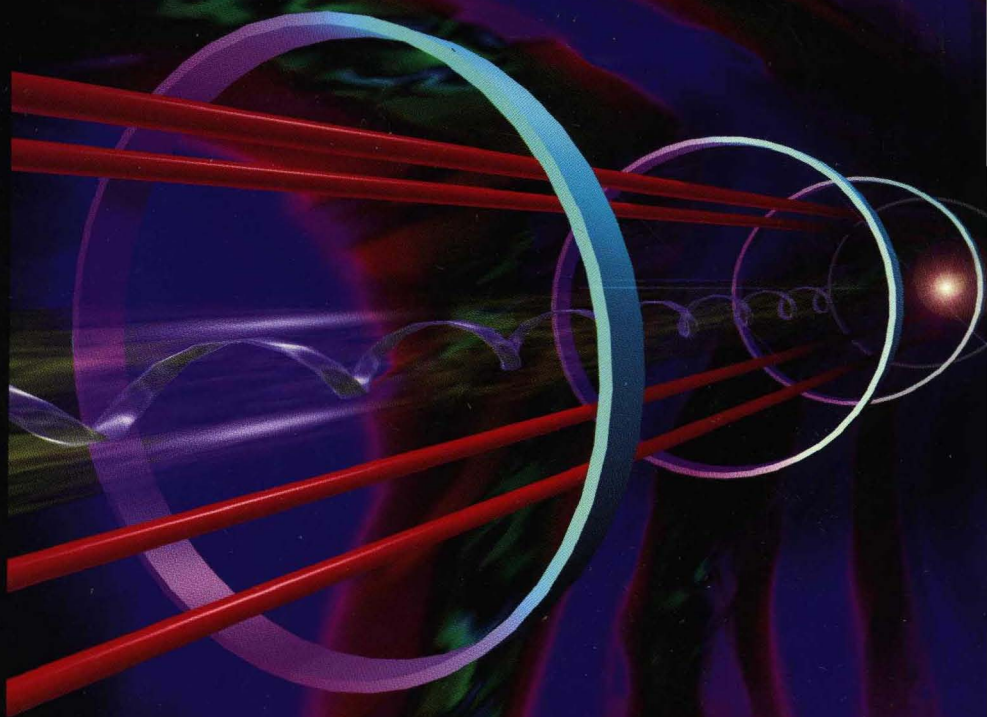
Vol. 2 1995

Flash Memory
Volume 2

PCMCIA Flash Memory Cards

PCMCIA-ATA Flash Memory Drives

Flash Automotive Components



WYLE *We're Customer Specific*

SANTA CLARA DIVISION
3000 Bowers Avenue
Santa Clara, CA 95051-0919
(408) 727-2500 • (800) 866-9953

intel

LITERATURE

For additional information on Intel products in the U.S. or Canada, call Intel's Literature Center at (800) 548-4725 or write to:

Intel Literature
P.O. Box 7641
Mt. Prospect, Il 60056-7641

To order literature outside of the U.S. and Canada contact your local international sales office.

CURRENT DATABOOKS

Product line databooks contain datasheets, application notes, article reprints, and other design information. Databooks can be ordered in the U.S. and Canada by calling TAB/McGraw-Hill at 1-800-822-8158; outside of the U.S. and Canada contact your local international sales office.

Title	Intel Order Number	ISBN
Automotive Products	231792	N/A
Embedded Applications (2 vol. set)	270648	1-55512-242-6
Embedded Microcontrollers	270646	1-55512-230-2
Embedded Microprocessors	272396	1-55512-231-0
Flash Memory (2 vol. set)	210830	1-55512-232-9
Intel486™ Microprocessors and Related Products	241731	1-55512-235-3
i960® Processors and Related Products	272084	1-55512-234-5
Military and Special Products	210461	N/A
Networking	297360	1-55512-236-1
OEM Boards, Systems and Software	280407	1-55512-237-X
Packaging	240800	1-55512-238-8
Pentium™ Processors and Related Products	241732	1-55512-239-6
Peripheral Components	296467	1-55512-240-X

A complete set of this information is available on CD-ROM through Intel's Data on Demand program, order number 240897. For information about Intel's Data on Demand ask for item number 240952.

24-HOUR AUTOMATED TECHNICAL SUPPORT*

Intel's Application Bulletin Board System (BBS) and FaxBack System are at your service, 24-hours a day, at no charge, and the information is updated frequently.

FaxBack SYSTEM

Technical and product information are available 24-hours a day! Order documents containing:

- Product Announcements
- Product Literature
- Intel Device Characteristics
- Design/Application Recommendations
- Stepping/Change Notifications
- Quality and Reliability Information

Information on the following subjects is also available:

- Microcontroller and Flash
- OEM Branded Systems
- Multibus/BBS Listing
- Multimedia
- Development Tools
- Quality and Reliability/Change Notification
- Microprocessor/PCI/Peripheral
- Intel Architecture Lab

To use FaxBack for Intel components and systems, dial **(800) 628-2283** or (916) 356-3105 (U.S. and Canada) *or +44{0} 1793-496646 (Europe)* and follow the automated voice-prompt menu. Document orders will be faxed to the fax number you specify. Catalogs are updated twice a month, so call for the latest information!

BULLETIN BOARD SYSTEM

Intel's Application Bulletin Board System (BBS) enables file retrieval 24-hours a day. The following can be located on the BBS:

- Software Drivers
- Tool Information
- Software/Application Utilities
- Product/Technical Documentation
- Firmware Upgrades
- Quality and Reliability Data

To use the Intel Application BBS (components and systems), dial **(916) 356-3600** for download access (U.S. and Canada) *or +44{0} 1793-496340 (Europe).* The BBS will support 1200–19200 baud rate modem. *Typical modem configuration: 9600 baud rate, No Parity, 8 Data Bits, 1 Stop Bit.* A directory listing of BBS files is also available through FaxBack or our 800 BBS (800-897-2536).

Retail Products

Information on Intel's retail products (Coprocessors and wireless, video, personal conferencing and network products) is available through the following services:

Internet : ftp.intel.com (143.185.65.2)
CompuServe : GO INTELFORUM (modem settings: E-7-1, up to 14.4 Kbps)

Country	BBS (N-8-1, up to 14.4 Kbps)	FaxBack
North America	(503) 264-7999	(800) 525-3019 or (503) 264-6835
Europe	+44 1 793-432955	+44 1 793-432509
Australia	+61 2 975-3066	+61 2 975-3922
Taiwan	+886 2 718-6422	+886 2 514-0815
Singapore	+65 256-4776	+65 256-5350
Hong Kong	+852 530-4116	+852 844-4448
Korea	+822 784-3430	+822 767-2594

*Support services provided courtesy of Intel Application Support

intel

Flash Memory
Volume 2

PCMCIA Flash Memory Cards, PCMCIA-ATA Flash Memory Drives, Flash Automotive Components

1995

Information in this document is provided solely to enable use of Intel products. Intel assumes no liability whatsoever, including infringement of any patent or copyright, for sale and use of Intel products except as provided in Intel's Terms and Conditions of Sale for such products.

Intel Corporation makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this document nor does it make a commitment to update the information contained herein.

Intel retains the right to make changes to these specifications at any time, without notice.

Contact your local Intel sales office or your distributor to obtain the latest specifications before placing your product order.

MDS is an ordering code only and is not used as a product name or trademark of Intel Corporation.

Intel Corporation and Intel's FASTPATH are not affiliated with Kinetics, a division of Excelan, Inc. or its FASTPATH trademark or products.

*Other brands and names are the property of their respective owners.

Additional copies of this document or other Intel literature may be obtained from:

Intel Corporation
Literature Sales
P.O. Box 7641
Mt. Prospect, IL 60056-7641

or call 1-800-879-4683

©INTEL CORPORATION, 1995

DATASHEET DESIGNATIONS

Intel uses various datasheet markings to designate each phase of the document as it relates to the product. The markings appear in the lower inside corner of each datasheet page. Following are the definitions of each marking:

Datasheet Marking	Description
Product Preview	Contains information on products in the design phase of development. Do not finalize a design with this information. Revised information will be published when the product becomes available.
Advanced Information	Contains information on products being sampled or in the initial production phase of development.*
Preliminary	Contains preliminary information on new products in production.*
No Marking	Contains information on products in full production.*

* Specifications within these datasheets are subject to change without notice. Verify with your local Intel sales office that you have the latest datasheet before finalizing a design.

Memory Overview 1

Flash Memory Overview and Support Tools 2

Synchronous & DRAM-Interface Flash Memory Components and FlashFile Components 3

Boot Block Components 4

Bulk-Erase Components 5

PCMCIA Flash Memory Cards 6

PCMCIA-ATA Flash Memory Drives 7

Flash Automotive Components 8

Process Engineering Reports 9

Flash Memory SmartDie™ Products 10

Article Reprints 11

Table of Contents

Alphanumeric Index xv

CHAPTER 1

Memory Overview

Intel Memory Technologies 1-1

CHAPTER 2

Flash Memory Overview and Support Tools

Flash Memory Overview 2-1

SUPPORT TOOLS

Flash Memory Customer Support 2-7
Sprint PLUS48/Intel Support Version 2-9
Multi-APRO Programmer/Intel Support Version 2-13
Flash Memory Card and Flash Drive Evaluation Kit Product Brief 2-16
Small Outline Package Guide Overview 2-18
Intel FLASHBuilder Software Product Brief 2-20

APPLICATION NOTES

AP-357 Power Supply Solutions for Flash Memory 2-22
AP-374 Flash Memory Write Protection Techniques 2-64
AB-29 Flash Memory Applications in Laser Printers 2-71

TECHNICAL PAPER

Small and Low Cost Power Supply Solution for Intel's Flash Memory Products 2-78

CHAPTER 3

Synchronous and DRAM-Interface Flash Memory Components

PRODUCT BRIEFS

Intel 28F016XS Embedded Flash RAM Product Brief 3-1
Intel 28F016XD Embedded Flash RAM Product Brief 3-2

DATA SHEETS

28F016XS 16-Mbit (1-Mbit x 16, 2-Mbit x 8) Synchronous Flash Memory 3-3
28F016XD 16-Mbit (1-Mbit x 16) DRAM - Interface Flash Memory 3-50

APPLICATION NOTES

AB-58 28F016XD - Based SIMM Designs 3-104
AP-398 Designing with the 28F016XS 3-112
AP-384 Designing with the 28F016XD 3-142
AP-600 Performance Benefits and Power/Energy Savings of 28F016XS-Based System Designs 3-172

TECHNICAL PAPERS

Interfacing the 28F016XS to the i960® Microprocessor Family 3-196
Interfacing the 28F016XS to the Intel486™ Microprocessor Family 3-238

FlashFile™ Components

DATA SHEETS

DD28F032SA 32-Mbit (2-Mbit x 16, 4-Mbit x 8) FlashFile Memory 3-257
28F016SV 16-Mbit (1-Mbit x 16, 2-Mbit x 8) FlashFile Memory 3-303
28F016SA 16-Mbit (1-Mbit x 16, 2-Mbit x 8) FlashFile Memory 3-359
Extended Temperature 28F016SA 16-Mbit (1-Mbit x 16, 2-Mbit x 8) FlashFile Memory 3-403
28F008SA 8-Mbit (1-Mbit x 8) FlashFile Memory *(Extended Temperature Specifications Included)* 3-443
28F008SA-L 8 Mbit (1Mbit x 8) Flash Memory 3-476

APPLICATION NOTES

AP-607 Multi-Site Layout Planning with Intel's FlashFile Components, Including ROM Compatibility 3-504

Table of Contents (Continued)

AP-393 28F016SV Compatibility with 28F016SA 3-533
AP-359 28F008SA Hardware Interfacing 3-551
AP-360 28F008SA Software Drivers 3-562
AP-362 Implementing Mobile PC Designs Using High Density FlashFile Components 3-585
AP-364 28F008SA Automation and Algorithms 3-640
AP-375 Upgrade Considerations from the 28F008SA to the 28F016SA 3-654
AP-377 16-Mbit Flash Product Family Software Drivers 28F016SA, 28F016SV, 28F016XS, 28F016XD 3-667
AP-378 System Optimization Using the Enhanced Features of the 28F016SA 3-722
AP-399 Implementing Mobile Intel486 SX Microprocessor PC Designs Using FlashFile™ Components 3-743
ENGINEERING REPORT
ER-27 The Intel 28F008SA Flash Memory 3-783

CHAPTER 4
Boot Block Components
DATA SHEETS
8-Mbit (512K x 16, 1-Mbit x 8) SmartVoltage Boot Block Flash Memory Family 4-1
4-Mbit (256K x 16, 512K x 8) SmartVoltage Boot Block Flash Memory Family *(Extended Temperature Specifications Included)* 4-13
2-Mbit (128K x 16, 256K x 8) SmartVoltage Boot Block Flash Memory Family *(Extended Temperature Specifications Included)* 4-68
28F400BX-T/B, 28F004BX-T/B 4 Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Family 4-124
28F400BL-T/B, 28F004BL-T/B 4-Mbit (256K x 16, 512K x 8) Low Power Boot Block Flash Memory Family 4-174
28F200BX-T/B, 28F002BX-T/B 2 Mbit (128K x 16, 256K x 8) Boot Block Flash Memory Family 4-218
28F200BL-T/B, 28F002BL-T/B 2-Mbit (128K x 16, 256K x 8) Low Power Boot Block Flash Memory Family 4-265
28F001BX-T/28F001BX-B 1M (128K x 8) CMOS Flash Memory 4-307
APPLICATION NOTES
AP-604 Using Intel's Boot Block Flash Memory Parameter Blocks to Replace EEPROM 4-341
AB-60 2/4/8-Mbit SmartVoltage Boot Block Flash Memory Family Overview 4-349
AP-341 Designing an Updatable BIOS Using Flash Memory 4-357
AP-363 Extended Flash Bios Concepts for Portable Computers 4-400
ENGINEERING REPORTS
ER-26 The Intel 28F001BX-T and 28F001BX-B Flash Memories 4-422
ER-29 The Intel 2/4-Mbit Boot Block Flash Memory Family 4-437
TECHNICAL PAPER
Boot Block Flash: The Next Generation White Paper 4-466

CHAPTER 5
Bulk-Erase Components
DATA SHEETS
28F020 2048K (256K x 8) CMOS Flash Memory 5-1
28F010 1024K (128K x 8) CMOS Flash Memory 5-33
28F512 512K (64K x 8) CMOS Flash Memory 5-64

Table of Contents (Continued)

28F256A 256K (32K x 8) CMOS Flash Memory 5-92
APPLICATION NOTES
AP-316 Using Flash Memory for In-System Reprogrammable Nonvolatile Storage . 5-117
AP-325 Guide to First Generation Flash Memory Programming 5-162
ENGINEERING REPORT
ER-24 Intel Flash Memory 28F256A, 28F512, 28F010, 28F020 5-184

CHAPTER 6
PCMCIA Flash Memory Cards
Series 2+ Flash Memory Card User's Manual Overview 6-1
DATA SHEETS
Series 2+ Flash Memory Cards 4-, 8-, 20- and 40-Megabyte 6-2
Series 2 Flash Memory Cards iMC002FLSA/iMC004FLSA/iMC010FLSA/ iMC020FLSA *(Extended Temperature Specifications Included)* 6-42
iMC004FLKA 4-Megabyte Flash Memory Card 6-80
iMC002FLKA 2-Mbyte Flash Memory Card 6-110
iMC001FLKA 1-Megabyte Flash Memory Card 6-140
APPLICATION NOTES
AP-606 Interchangeability of Series 1, Series 2 and Series 2+ Flash Memory Cards 6-170
AP-343 Solutions for High Density Applications Using Intel Flash Memory 6-185
AP-361 Implementing the Integrated Registers of the Series 2 Flash Memory Card . 6-215
SUPPORT TOOLS
Flash Memory Card and Flash Drive Evaluation Kit Product Brief 6-232
TECHNICAL PAPER
Intel FlashFile Memory-The Key to Diskless Mobile PCs 6-234

CHAPTER 7
PCMCIA-ATA Flash Memory Drives
DATA SHEET
iFD005P2SA/iFD010P2SA Flash Drive 7-1
DESIGN GUIDE
Flash Drive Design Guide (for iFD005P2SA/iFD010P2SA) 7-46
SUPPORT TOOLS
Flash Memory Card and Flash Drive Evaluation Kit Product Brief 7-118

CHAPTER 8
Flash Automotive Components
DATA SHEETS
A28F400BR-T/B 4-Mbit (256K x 16, 512K x 8) SmartVoltage Boot Block Flash Memory Family 8-1
A28F400BX-T/B 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Family... 8-35
A28F200BR-T/B 2-Mbit (128K x 16, 256K x 8) SmartVoltage Boot Block Flash Memory Family 8-69
A28F200BX-T/B 2-Mbit (128K x 16, 256K x 8) Boot Block Flash Memory Family... 8-102
A28F010 1024K (128K x 8) CMOS Flash Memory 8-135
A28F512 512K (64K x 8) CMOS Flash Memory 8-158
A28F256A 256K (32K x 8) CMOS Flash Memory 8-182

CHAPTER 9
Process Engineering Reports
ER-20 ETOX II Flash Memory Technology 9-1
ER-28 ETOX III Flash Memory Technology 9-6

Table of Contents (Continued)

ER-33 ETOX IV Flash Memory Technology: Insight to Intel's Fourth Generation Process Innovation 9-19

CHAPTER 10

Flash Memory SmartDie™ Products

DATA SHEETS

28F010 1024K (128K x 8) Flash Memory 10-1
28F020 2048K (256K x 8) Flash Memory 10-9
28F008SA 8 Mbit (1 Mbit x 8) Flash Memory 10-17
28F400BX-T 4 Mbit (256K x 16, 512K x 8) Boot Block Flash Memory 10-25
28F001BX-T/B 1M (128K x 8) CMOS Flash Memory 10-33

CHAPTER 11

Article Reprints

AR-710 Flash Solid-State Drive with 6 MBs Read/Write Channel and Data Compression 11-1
AR-711 Flash: Big New in Storage? 11-4
AR-715 Flash Memory: Meeting the Needs of Mobile Computing 11-8
AR-716 Flash Memory for Top Speeds in Mobile Computing Applications 11-16
AR-717 The Many Facts of Flash Memory 11-18
AR-718 Standardizing on a Flash File System 11-28
AR-723 Interfacing BootBlock Flash Memories to the MCS 96 Family 11-32

Alphanumeric Index

28F001BX-T/28F001BX-B 1M (128K x 8) CMOS Flash Memory 4-307
28F001BX-T/B 1M (128K x 8) CMOS Flash Memory 10-33
28F008SA 8 Mbit (1 Mbit x 8) Flash Memory 10-17
28F008SA 8-Mbit (1-Mbit x 8) FlashFile™ Memory *(Extended Temperature Specifications Included)* 3-443
28F008SA-L 8 Mbit (1Mbit x 8) Flash Memory 3-476
28F010 1024K (128K x 8) CMOS Flash Memory 5-33
28F010 1024K (128K x 8) Flash Memory 10-1
28F016SA 16-Mbit (1-Mbit x 16, 2-Mbit x 8) FlashFile Memory 3-359
28F016SV 16-Mbit (1-Mbit x 16, 2-Mbit x 8) FlashFile Memory 3-303
28F016XD 16-Mbit (1-Mbit x 16) DRAM - Interface Flash Memory 3-50
28F016XS 16-Mbit (1-Mbit x 16, 2-Mbit x 8) Synchronous Flash Memory 3-3
28F020 2048K (256K x 8) CMOS Flash Memory 5-1
28F020 2048K (256K x 8) Flash Memory 10-9
28F200BL-T/B, 28F002BL-T/B 2-Mbit (128K x 16, 256K x 8) Low Power Boot Block Flash Memory Family 4-265
28F200BX-T/B, 28F002BX-T/B 2 Mbit (128K x 16, 256K x 8) Boot Block Flash Memory Family 4-218
28F256A 256K (32K x 8) CMOS Flash Memory 5-92
28F400BL-T/B, 28F004BL-T/B 4-Mbit (256K x 16, 512K x 8) Low Power Boot Block Flash Memory Family 4-174
28F400BX-T 4 Mbit (256K x 16, 512K x 8) Boot Block Flash Memory 10-25
28F400BX-T/B, 28F004BX-T/B 4 Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Family 4-124
28F512 512K (64K x 8) CMOS Flash Memory 5-64
2-Mbit (128K x 16, 256K x 8) SmartVoltage Boot Block Flash Memory Family *(Extended Temperature Specifications Included)* 4-68
4-Mbit (256K x 16, 512K x 8) SmartVoltage Boot Block Flash Memory Family *(Extended Temperature Specifications Included)* 4-13
8-Mbit (512K x 16, 1-Mbit x 8) SmartVoltage Boot Block Flash Memory Family 4-1
A28F010 1024K (128K x 8) CMOS Flash Memory 8-135
A28F200BR-T/B 2-Mbit (128K x 16, 256K x 8) SmartVoltage Boot Block Flash Memory Family 8-69
A28F200BX-T/B 2-Mbit (128K x 16, 256K x 8) Boot Block Flash Memory Family 8-102
A28F256A 256K (32K x 8) CMOS Flash Memory 8-182
A28F400BR-T/B 4-Mbit (256K x 16, 512K x 8) SmartVoltage Boot Block Flash Memory Family 8-1
A28F400BX-T/B 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Family 8-35
A28F512 512K (64K x 8) CMOS Flash Memory 8-158
AB-29 Flash Memory Applications in Laser Printers 2-71
AB-58 28F016XD - Based SIMM Designs 3-104
AB-60 2/4/8-Mbit SmartVoltage Boot Block Flash Memory Family Overview 4-349
AP-316 Using Flash Memory for In-System Reprogrammable Nonvolatile Storage 5-117
AP-325 Guide to First Generation Flash Memory Programming 5-162
AP-341 Designing an Updatable BIOS Using Flash Memory 4-357
AP-343 Solutions for High Density Applications Using Intel Flash Memory 6-185
AP-357 Power Supply Solutions for Flash Memory 2-22
AP-359 28F008SA Hardware Interfacing 3-551
AP-360 28F008SA Software Drivers 3-562
AP-361 Implementing the Integrated Registers of the Series 2 Flash Memory Card 6-215
AP-362 Implementing Mobile PC Designs Using High Density FlashFile Components 3-585
AP-363 Extended Flash Bios Concepts for Portable Computers 4-400

Alphanumeric Index (Continued)

AP-364 28F008SA Automation and Algorithms 3-640
AP-374 Flash Memory Write Protection Techniques 2-64
AP-375 Upgrade Considerations from the 28F008SA to the 28F016SA 3-654
AP-377 16-Mbit Flash Product Family Software Drivers 28F016SA, 28F016SV, 28F016XS, 28F016XD 3-667
AP-378 System Optimization Using the Enhanced Features of the 28F016SA 3-722
AP-384 Designing with the 28F016XD 3-142
AP-393 28F016SV Compatibility with 28F016SA 3-533
AP-398 Designing with the 28F016XS 3-112
AP-399 Implementing Mobile Intel486™ SX Microprocessor PC Designs Using FlashFile Components 3-743
AP-600 Performance Benefits and Power/Energy Savings of 28F016XS-Based System Designs 3-172
AP-604 Using Intel's Boot Block Flash Memory Parameter Blocks to Replace EEPROM ... 4-341
AP-606 Interchangeability of Series 1, Series 2 and Series 2+ Flash Memory Cards 6-170
AP-607 Multi-Site Layout Planning with Intel's FlashFile Components, Including ROM Compatibility 3-504
AR-710 Flash Solid-State Drive with 6 MBs Read/Write Channel and Data Compression .. 11-1
AR-711 Flash: Big New in Storage? 11-4
AR-715 Flash Memory: Meeting the Needs of Mobile Computing 11-8
AR-716 Flash Memory for Top Speeds in Mobile Computing Applications 11-16
AR-717 The Many Facts of Flash Memory 11-18
AR-718 Standardizing on a Flash File System 11-28
AR-723 Interfacing BootBlock Flash Memories to the MCS 96 Family 11-32
Boot Block Flash: The Next Generation White Paper 4-466
DD28F032SA 32-Mbit (2-Mbit x 16, 4-Mbit x 8) FlashFile Memory 3-257
ER-20 ETOX II Flash Memory Technology 9-1
ER-24 Intel Flash Memory 28F256A, 28F512, 28F010, 28F020 5-184
ER-26 The Intel 28F001BX-T and 28F001BX-B Flash Memories 4-422
ER-27 The Intel 28F008SA Flash Memory 3-783
ER-28 ETOX III Flash Memory Technology 9-6
ER-29 The Intel 2/4-Mbit Boot Block Flash Memory Family 4-437
ER-33 ETOX IV Flash Memory Technology: Insight to Intel's Fourth Generation Process Innovation 9-19
Extended Temperature 28F016SA 16-Mbit (1-Mbit x 16, 2-Mbit x 8) FlashFile Memory 3-403
Flash Drive Design Guide (for iFD005P2SA/iFD010P2SA) 7-46
Flash Memory Card and Flash Drive Evaluation Kit Product Brief 7-118
Flash Memory Card and Flash Drive Evaluation Kit Product Brief 6-232
Flash Memory Card and Flash Drive Evaluation Kit Product Brief 2-16
Flash Memory Customer Support 2-7
Flash Memory Overview 2-1
iFD005P2SA/iFD010P2SA Flash Drive 7-1
iMC001FLKA 1-Megabyte Flash Memory Card 6-140
iMC002FLKA 2-Mbyte Flash Memory Card 6-110
iMC004FLKA 4-Megabyte Flash Memory Card 6-80
Intel 28F016XD Embedded Flash RAM Product Brief 3-2
Intel 28F016XS Embedded Flash RAM Product Brief 3-1
Intel FLASHBuilder Software Product Brief 2-20
Intel FlashFile Memory-The Key to Diskless Mobile PCs 6-234
Intel Memory Technologies 1-1
Interfacing the 28F016XS to the Intel486 Microprocessor Family 3-238
Interfacing the 28F016XS to the i960® Microprocessor Family 3-196

Alphanumeric Index (Continued)

Multi-APRO Programmer/Intel Support Version 2-13
Series 2 Flash Memory Cards iMC002FLSA/iMC004FLSA/iMC010FLSA/iMC020FLSA
(Extended Temperature Specifications Included) 6-42
Series 2+ Flash Memory Card User's Manual Overview 6-1
Series 2+ Flash Memory Cards 4-, 8-, 20- and 40-Megabyte 6-2
Small and Low Cost Power Supply Solution for Intel's Flash Memory Products 2-78
Small Outline Package Guide Overview 2-18
Sprint PLUS48/Intel Support Version 2-9

intel®

6

PCMCIA Flash Memory Cards

SERIES 2+ FLASH MEMORY CARD USER'S MANUAL OVERVIEW

The documentation for Intel's Series 2+ Flash Memory Card includes the **SERIES 2+ FLASH MEMORY CARD USER'S MANUAL** (297373) and the Series **2+ Datasheet** (290491). The user's manual provides a complete description of the methods for using the Series 2+ Flash Memory Card, from both a hardware and a software perspective. The data sheet provides all AC and DC characteristics (including timing waveforms) and a convenient reference for the device command set and the card's integrated registers.

The user's guide encompasses all aspects of the Series 2+ Card in detail, ranging from the 28F016SA to the card's PCMCIA electrical interface. Each subject discussed within this document is organized to minimize the OEM's efforts for comprehending the functionality of the Series 2+ card.

SERIES 2+ USER'S MANUAL: CHAPTER OVERVIEW

Chapter 2: Card Architecture

This chapter discusses the hardware aspects of the Series 2+ Card, including its integrated V_{PP} generator, Smart-Power circuitry, Component Management Registers, PCMCIA interface, and device layout.

Chapter 3: Device Architecture

This chapter explains the basic functionality of the 28F016SA, the integral foundation for the Series 2+ Card. In particular, the device status registers are thoroughly discussed—down to the bit level.

Chapter 4: Command Descriptions

This chapter, supported by the algorithms in Chapter 8, discusses all the details of every command the 28F016SA can execute. The chapter is divided into two major sections: the 28F008SA-compatible commands and the new Performance Enhancement commands.

Chapter 5: Application Programming Specifics

This chapter goes into more detail on some of the special capabilities of the Series 2+ Card. This includes page buffer operation, data protection, ready/busy modes, and power conservation.

Chapter 6: 40-Megabyte Series 2+ Card Operation

This chapter explains the difference, from a programmer's standpoint, between the 4- or 20-megabyte cards (28F016SA-based) and the 40-megabyte card (28F032SA-based).

Chapter 7: Series 2 Compatibility

The Series 2+ Card was designed to be backwards-compatible with the Series 2 Card. However, a few minor differences exist due to the advancement in the technology. This chapter explains those differences and how to accommodate them.

Chapter 8: Software Algorithms

This chapter supports all the other sections in this user's guide dealing with software programming. It includes flowcharts for the 28F016SA commands as well as C and Assembly code for some of the basic routines.

SERIES 2+ FLASH MEMORY CARDS
4-, 8-, 20- AND 40-MEGABYTE

iMC004FLSP, iMC008FLSP, iMC020FLSP, iMC040FLSP

- **Single Power Supply**
- **Automatically Reconfigure for 3.3V and 5V Systems**
- **150 ns Maximum Access Time with 5V Power Supply**
- **250 ns Maximum Access Time with 3.3V Power Supply**
- **High-Performance Random Writes**
  - **— 0.85 MB/S Sustained Throughput**
  - **— 1 KB Burst Write at 10 MB/S**
- **12 μA Typical Deep Power-Down**
- **Revolutionary Architecture**
  - **— Pipelined Command Execution**
  - **— Write during Erase**
  - **— Series 2 Command Super-Set**
- **State-of-the-Art 0.6 μm ETOX™ IV Flash Technology**
- **1 Million Erase Cycles per Block**
- **Up to 640 Independent Lockable Blocks**
- **PCMCIA 2.1/JEIDA 4.1-Compatible**
- **PCMCIA Type 1 Form Factor**
- **Series 2+ User's Manual**

Intel's Series 2+ Flash Memory Card sets the new record for high-performance disk emulation and eXecute-In-Place (XIP) applications in mobile PCs and dedicated equipment. Manufactured with Intel's 28F016SA 16-Mbit (DD28F032SA 32-Mbit) FlashFile™ Memory, this card takes advantage of a revolutionary architecture that provides innovative capabilities, low-power operation and very high read/write performance.

The Series 2+ Card provides today's highest density, highest performance nonvolatile read/write solution for solid-state storage applications. These applications are enhanced further with this product's symmetrically blocked architecture, extended MTBF, low-power 3.3V operation, built-in V_{PP} generator, and multiple block-locking methods. The Series 2+ Card's dual read and write voltages allow interchange between 3.3V and 5.0V systems.

290491-8

ETOX and FlashFile™ are trademarks of Intel Corporation.

November 1994
Order Number: 290491-006

Series 2+ Flash Memory Cards

CONTENTS

PAGE

1.0 SCOPE OF DOCUMENT 6-4

2.0 PRODUCT OVERVIEW 6-4

3.0 SERIES 2+ ARCHITECTURE OVERVIEW 6-5

3.1 Card Pinout and Pin Description 6-5

3.2 Series 2+ Card Control Logic 6-10

3.3 Component Management Registers 6-11

3.4 SmartPower 6-11

4.0 DEVICE COMMAND SET 6-14

5.0 DEVICE STATUS REGISTER 6-17

6.0 PCMCIA CARD INFORMATION STRUCTURE 6-18

7.0 SYSTEM DESIGN CONSIDERATIONS 6-22

7.1 Power Supply Decoupling 6-22

7.2 Power-Up/Down Protection 6-22

7.3 Hot Insertion/Removal 6-22

8.0 ELECTRICAL SPECIFICATIONS 6-23

8.1 Absolute Maximum Ratings 6-23

9.0 SERIES 2+ DC CHARACTERISTICS 6-24

9.1 General 6-24

9.2 CMOS Interfacing: V_{CC} = 3.3V ... 6-25

9.3 CMOS Interfacing: V_{CC} = 5.0V ... 6-26

9.4 TTL Interfacing: V_{CC} = 3.3V 6-27

9.5 TTL Interfacing: V_{CC} = 5.0V 6-28

CONTENTS

PAGE

10.0 AC CHARACTERISTICS 6-29

10.1 Read Operations: Common and Attribute Memory 6-29

10.2 Write Operations: Common and Attribute Memory 6-31

10.3 CE#-Controlled Write Operations: Common and Attribute Memory 6-33

10.4 Power-Up/Power-Down 6-35

10.5 Series 2+ Card Information Structure Tuple Changes 6-36

11.0 CAPACITANCE 6-36

12.0 ERASE AND DATA WRITE PERFORMANCE 6-37

13.0 PACKAGING 6-38

14.0 ADDITIONAL INFORMATION 6-40

14.1 Ordering Information 6-40

14.2 References 6-40

14.3 Revision History 6-41

1.0 SCOPE OF DOCUMENT

The documentation for Intel's Series 2+ Flash Memory Card includes this datasheet and the ***Series 2+ Flash Memory Card User's Manual*** (297373). The datasheet provides all AC and DC characteristics (including timing waveforms) and a convenient reference for the device command set and the card's integrated registers (including the 28F016SA's status registers). The ***Series 2+ Memory Card User's Manual*** provides a complete description of the methods for using the card. It also contains the full list of software algorithms and flowcharts and a section for upgrading Intel's Series 2 Flash Memory Cards designs.

2.0 PRODUCT OVERVIEW

The 4-, 8-, and 20- Megabyte Series 2+ Flash Memory Cards each contain a flash memory array that consists of two, four, and ten 28F016SA TSOP memory devices, respectively. Each 28F016SA contains 32 distinct, individually-erasable, 64-Kbyte blocks. Therefore, the 4-, 8-, and 20-megabyte cards contain 64, 128 and 320 independantly lockable blocks, respectively.

The 40-Megabyte Series 2+ Flash Memory Cards contain a flash memory array that consists of ten DD28F032SA TSOP memory devices. Each DD28F032SA contains two 28F016SA die in a single package, resulting in 64 distinct, individually-erasable, 64-Kbyte blocks. The 40-megabyte cards have 640 independently lockable blocks.

The Series 2+ Card offers additional product features to those of the Series 2 Card family (refer to the iMC0XXFLSA datasheets). Some of the more notable card-level enhancements include: interchangeable operation at 3.3V or 5.0V, block locking and internal V_{PP} generation.

The Series 2+ card incorporates V_{CC} detect circuitry, referred to as SmartPower, to sense the voltage level present at the card interface. The card's control logic automatically configures its circuitry and the 28F016SA/DD28F032SA memory array accordingly. The Card Information Structure (CIS) reports that the card is 3.3V or 5.0V compatible. The card also detects the presence of 12.0V on the V_{PP} pin and passes this supply to each memory device. When the 12.0V power supply is unavailable, the card can generate the required V_{PP} via its internal V_{PP} generation circuitry, whether V_{CC} is 3.3V or 5.0V.

At the device level, internal algorithm automation allows Write and Erase operations to be executed using a two-write command sequence in the same way as the 28F008SA FlashFile memory in the Series 2 Card. A super-set of commands and additional performance enhancements have been added to the basic 28F008SA command set:

- Page Buffer Write to Flash results in writes up to four times faster than Series 2 Cards.
- Command Queueing permits the devices to receive new commands during the execution of the current command set.
- Automatic data writes during erase allows the 28F016SA to perform Write operations to one block of memory while performing an Erase on another block.
- Software locking of memory blocks provides a means to selectively protect code or data within the card.
- Erase all unlocked blocks provides a quick and simple method to sequentially erase all the blocks within a 28F016SA memory device.

The Series 2+ Card has two ways to put the flash devices into a sleep mode for reduced power consumption:

1. Issue a command to individual devices, referred to as the software-controlled sleep mode. The device will retain status register data contents and finish any operation in progress using this approach.
2. Write to the card's PCMCIA-compatible Configuration and Status Register to activate a Reset Power-Down to all devices simultaneously.

The card achieves its PCMCIA-compatible word-wide access by pairing the 28F016SA/DD28F032SA devices resulting in an accessible memory block size of 64 Kwords. The card's decoding logic (contained within the ASICs) allows the system to write or read one word at a time, or one byte at a time by referencing the high or low byte. Erasure can be performed on the entire block pair (high and low byte simultaneously) or on the high and low portions separately. Although the 28F016SA/DD28F032SA support byte or word-wide data access, the byte interface was utilized within the card to allow the delivery of higher performance benefits, such as doubling the effective page buffer size and write performance.

The Series 2+ Card's ASICs also contain the Component Management Registers that provide five control functions: Ready-Busy Mode Selection, Software Write Protection, Card Status, Voltage Control, and Soft Reset.

The memory card interface supports the Personal Computer Memory Card Industry Association (PCMCIA 2.10) and Japanese Electronics Industry Development Association (JEIDA 4.1) 68-pin card format. The Series 2+ Flash Card meets all PCMCIA/JEIDA Type 1 mechanical specifications.

3.0 SERIES 2+ ARCHITECTURE OVERVIEW

The Series 2+ Card consists of three major functional elements—the flash memory array, card control and SmartPower circuitry. The card control logic handles the interface between the flash memory array and the host system's PCMCIA signals. SmartPower circuitry provides the card's integrated V_{PP} generator and a means for detecting the socket's voltage levels.

3.1 Card Pinout and Pin Description

The 68-pin PCMCIA format provides the system interface for the Series 2+ Flash Memory Card (see Tables 1 and 2). The detailed specifications for this interface is described in the PCMCIA 2.10 Standard Specification. The Series 2+ Flash Card product family conforms to the requirements of previous PCMCIA Versions Release 1.0, Release 2.0 and Release 2.01 of the PC Card Standard. Release 2.10 redefined pins 43 and 57 as VS_1 and VS_2 (previously REFRESH and RFU, respectively).

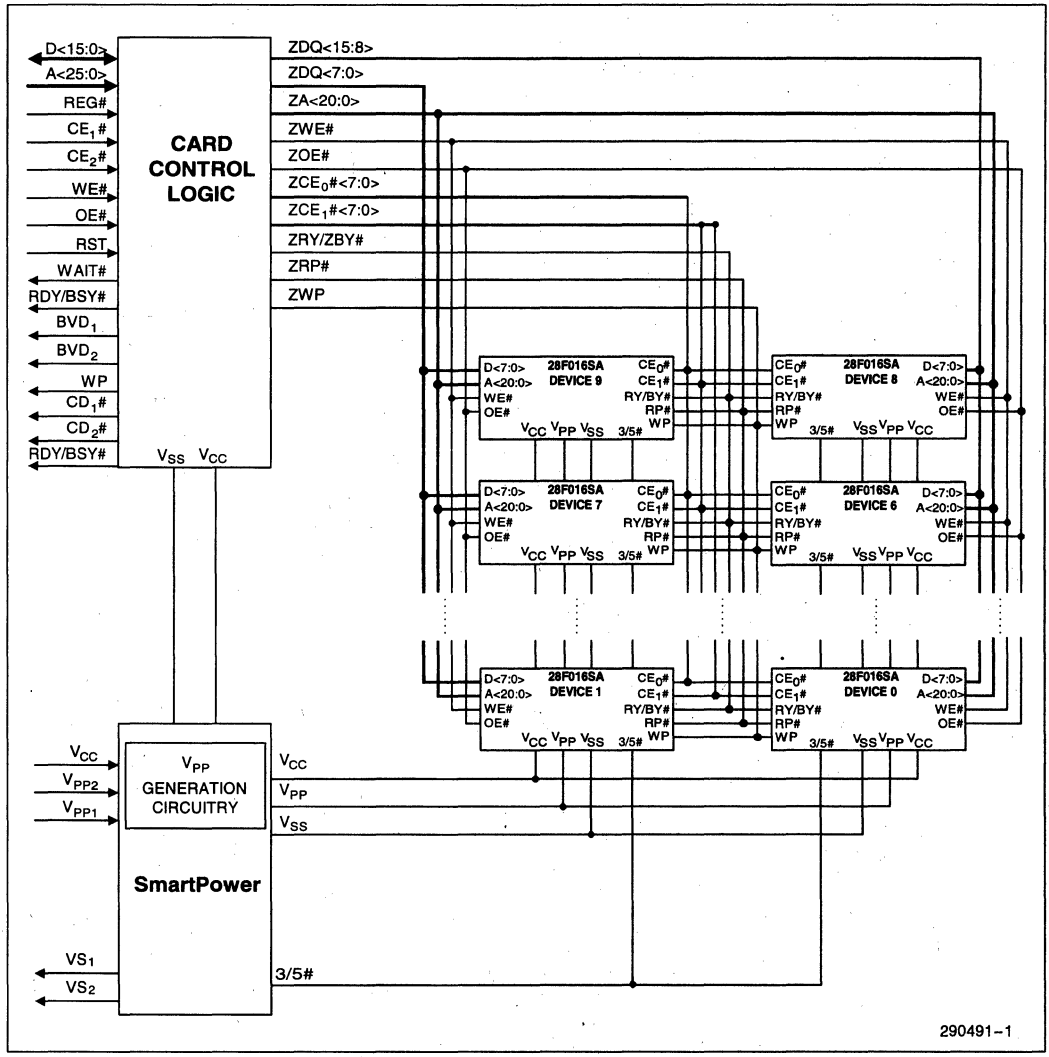


Figure 1. Series 2+ Flash Memory Card Block Diagram Showing Major Functional Elements

Table 1. Series 2+ Flash Memory Card Signals

Pin	Signal	I/O	Function	Active
1	GND		Ground	
2	DQ_3	I/O	Data Bit 3	
3	DQ_4	I/O	Data Bit 4	
4	DQ_5	I/O	Data Bit 5	
5	DQ_6	I/O	Data Bit 6	
6	DQ_7	I/O	Data Bit 7	
7	CE_1#	I	Card Enable 1	LOW
8	A_{10}	I	Address Bit 10	
9	OE#	I	Output Enable	LOW
10	A_{11}	I	Address Bit 11	
11	A_9	I	Address Bit 9	
12	A_8	I	Address Bit 8	
13	A_{13}	I	Address Bit 13	
14	A_{14}	I	Address Bit 14	
15	WE#	I	Write Enable	LOW
16	RDY/BSY#	O	Ready/Busy	LOW
17	V_{CC}		Supply Voltage	
18	V_{PP1}		Supply Voltage	
19	A_{16}	I	Address Bit 16	
20	A_{15}	I	Address Bit 15	
21	A_{12}	I	Address Bit 12	
22	A_7	I	Address Bit 7	
23	A_6	I	Address Bit 6	
24	A_5	I	Address Bit 5	
25	A_4	I	Address Bit 4	
26	A_3	I	Address Bit 3	

Pin	Signal	I/O	Function	Active
27	A_2	I	Address Bit 2	
28	A_1	I	Address Bit 1	
29	A_0	I	Address Bit 0	
30	DQ_0	I/O	Data Bit 0	
31	DQ_1	I/O	Data Bit 1	
32	DQ_2	I/O	Data Bit 2	
33	WP	O	Write Protect	HIGH
34	GND		Ground	
35	GND		Ground	
36	CD_1#	O	Card Detect 1	LOW
37	DQ_{11}	I/O	Data Bit 11	
38	DQ_{12}	I/O	Data Bit 12	
39	DQ_{13}	I/O	Data Bit 13	
40	DQ_{14}	I/O	Data Bit 14	
41	DQ_{15}	I/O	Data Bit 15	
42	CE_2#	I	Card Enable 2	LOW
43	VS_1	O	Voltage Sense 1	LOW
44	RFU		Reserved	
45	RFU		Reserved	
46	A_{17}	I	Address Bit 17	
47	A_{18}	I	Address Bit 18	
48	A_{19}	I	Address Bit 19	
49	A_{20}	I	Address Bit 20	
50	A_{21}	I	Address Bit 21	
51	V_{CC}		Supply Voltage	
52	V_{PP2}		Supply Voltage	

Table 1. Series 2+ Flash Memory Card Signals (Continued)

Pin	Signal	I/O	Function	Active
53	A_{22}	I	Address Bit 22	
54	A_{23}	I	Address Bit 23	
55	A_{24}	I	Address Bit 24	
56	A_{25}	I	Address Bit 25	
57	VS_2	O	Voltage Sense 2	N.C.
58	RST	I	Reset	HIGH
59	WAIT#	O	Extend Bus Cycle	LOW
60	RFU		Reserved	

Pin	Signal	I/O	Function	Active
61	REG#	I	Attribute Memory Select	LOW
62	BVD_2	O	Battery Voltage Detect 2	
63	BVD_1	O	Battery Voltage Detect 1	
64	DQ_8	I/O	Data Bit 8	
65	DQ_9	I/O	Data Bit 9	
66	DQ_{10}	I/O	Data Bit 10	
67	CD_2#	O	Card Detect 2	LOW
68	GND		Ground	

Table 2. Series 2+ Flash Memory Card Signal Description

Symbol	Type	Name and Function
A_0–A_{25}	I	**ADDRESS INPUTS:** Address A_0 through A_{25} are address bus lines which enable direct addressing of up to 64 megabytes of memory on the card. Signal A_0 is not used in word access mode. A_{25} is the most significant bit.
DQ_0–DQ_{15}	I/O	**DATA INPUT/OUTPUT:** DQ_0 through DQ_{15} constitute the bi-directional data bus. DQ_{15} is the most significant bit.
CE_1#,CE_2#	I	**CARD ENABLE 1 & 2:** CE_1# enables even bytes, CE_2# enables odd bytes. Multiplexing A_0, CE_1# and CE_2# allows 8-bit hosts to access all data on D_0 through D_7.
OE#	I	**OUTPUT ENABLE:** Active low signal gating read data from the memory card.
WE#	I	**WRITE ENABLE:** Active low signal gating write data to the memory card.
RDY/BSY#	O	**READY/BUSY OUTPUT:** Indicates status of internally timed erase or write activities. A high output indicates the memory card is ready to accept accesses. A low output indicates that a device in the memory card is busy with internally timed erase or write activities.
CD_1#,CD_2#	O	**CARD DETECT 1 & 2:** These signals provide for correct memory card insertion detection. They are positioned at opposite ends of the card to detect proper alignment. The signals are connected to ground internally on the memory card, and will be forced low whenever a card is placed in the socket. The host socket interface circuitry shall supply 10K or larger pull-up resistors on these signal pins.
WP	O	**WRITE PROTECT:** Write Protect reflects the status of the Write Protect switch on the memory card. WP set to high = write protected, providing internal hardware write lockout to the flash array.
V_{PP1},V_{PP2}		**WRITE/ERASE POWER SUPPLY:** (12V nominal) for erasing memory array blocks or writing bytes in the array. These pins must be 12V to perform and Write/Erase operation when not using the card's integrated V_{PP} generator. These signals may be disconnected but are required for ExCA™ standard compliance.
V_{CC}		**CARD POWER SUPPLY:** (3.3V or 5V nominal) for all internal circuitry.
GND		**GROUND** for all internal circuitry.
REG#	I	**REGISTER SELECT:** Provides access to Series 2+ Flash Memory Card registers and Card Information Structure in the Attribute Memory Plane.
RST	I	**RESET:** Active high signal for placing card in Power-On Default State.
WAIT#	O	**WAIT:** (Extend Bus Cycle) This signal is driven high for compatibility.
BVD_1, BVD_2	O	**BATTERY VOLTAGE DETECT:** These signals are driven high to maintain SRAM card compatibility.
VS_1, VS_2	O	**VOLTAGE SENSE:** Notify the host socket of the card's V_{CC} requirements. VS_1 grounded and VS_2 open indicates a 3.3V/5V card has been inserted.
RFU		**RESERVED FOR FUTURE USE**
N.C.		**NO INTERNAL CONNECTION TO CARD;** pin may be driven or left floating.

3.2 Series 2+ Card Control Logic

The Card Control Logic, contained within two ASICS, handles the address decoding and data control for the Series 2+ Card. The Component Management Registers are also contained within the Card Control Logic.

3.2.1 ADDRESS DECODE LOGIC

At the highest level, the Address Decode section determines when to select the Common Memory (REG# = V_{IH}) or Attribute Memory (REG# = V_{IL}) Planes. Within the Attribute Memory Plane (Figure 2) , the address decode logic determines when to select the Card Information Structure (CIS) or Component Management Registers (CMR). The CIS contains tuple information and is located at even-byte addresses beginning with address 0000H (refer to Section 6.0). The Component Management Registers are mapped at even byte locations beginning at address 4000H (refer to Section 3.3 for a detailed description).

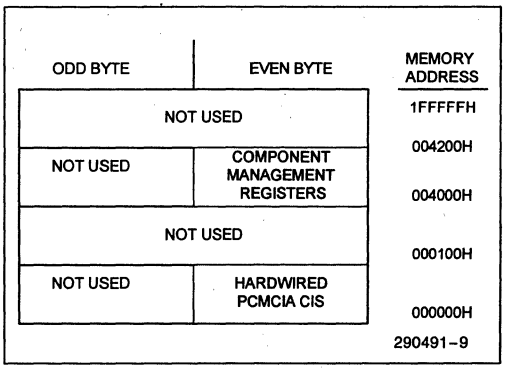


Figure 2. Attribute Memory Plane

3.2.2 DATA CONTROL

As shown in Table 3, data paths and directions are selected by the Data Control logic using REG#, A_0, WE#, OE#, CE_1#, and CE_2# as logic inputs. The Data Control logic selects any of the PCMCIA word-wide, byte-wide, and odd-byte modes for either Reads or Writes to Common or Attribute Memory. All accesses to the Attribute Memory Plane must be made through D[7:0] no valid data can be written on the high byte. Reads of D[15:8] will yield FFH.

Table 3. Data Access Mode Truth Table

COMMON MEMORY PLANE										
Mode	**REG#**	**CE_2#**	**CE_1#**	**A_0**	**OE#**	**WE#**	**V_{PP2}**	**V_{PP1}**	**D[15:8]**	**D[7:0]**
Standby	X	V_{IH}	V_{IH}	X	X	X	V_{PPL}	V_{PPL}	High-Z	High-Z
Byte-Read	V_{IH}	V_{IH}	V_{IL}	V_{IL}	V_{IL}	V_{IH}	V_{PPL}	V_{PPL}	High-Z	Even
	V_{IH}	V_{IH}	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{PPL}	V_{PPL}	High-Z	Odd
Word-Read	V_{IH}	V_{IL}	V_{IL}	X	V_{IL}	V_{IH}	V_{PPL}	V_{PPL}	Odd	Even
Odd Byte-Read	V_{IH}	V_{IL}	V_{IH}	X	V_{IL}	V_{IH}	V_{PPL}	V_{PPL}	Odd	High-Z
Byte-Write	V_{IH}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{IL}	XXX	V_{PPH}	XXX	Even
	V_{IH}	V_{IH}	V_{IL}	V_{IH}	V_{IH}	V_{IL}	V_{PPH}	XXX	XXX	Odd
Word-Write	V_{IH}	V_{IL}	V_{IL}	X	V_{IH}	V_{IL}	V_{PPH}	V_{PPH}	Odd	Even
Odd Byte-Write	V_{IH}	V_{IL}	V_{IH}	X	V_{IH}	V_{IL}	V_{PPH}	V_{PPL}	Odd	XXX

Table 3. Data Access Mode Truth Table (Continued)

ATTRIBUTE MEMORY PLANE										
Mode	**REG#**	**CE_2#**	**CE_1#**	**A_0**	**OE#**	**WE#**	**V_{PP2}**	**V_{PP1}**	**D[15:8]**	**D[7:0]**
Standby	X	V_{IH}	V_{IH}	X	X	X	V_{PPL}	V_{PPL}	High-Z	High-Z
Byte-Read	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IL}	V_{IH}	V_{PPL}	V_{PPL}	High-Z	Even
	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{PPL}	V_{PPL}	High-Z	FFH
Word-Read	V_{IL}	V_{IL}	V_{IL}	X	V_{IL}	V_{IH}	V_{PPL}	V_{PPL}	FFH	Even
Odd Byte-Read	V_{IL}	V_{IL}	V_{IH}	X	V_{IL}	V_{IH}	V_{PPL}	V_{PPL}	FFH	High-Z
Byte-Write	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{IL}	V_{PPL}	V_{PPL}	XXX	Even
	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{IH}	V_{IL}	V_{PPL}	V_{PPL}	XXX	XXX
Word-Write	V_{IL}	V_{IL}	V_{IL}	X	V_{IH}	V_{IL}	V_{PPL}	V_{PPL}	XXX	Even
Odd Byte-Write	V_{IL}	V_{IL}	V_{IH}	X	V_{IH}	V_{IL}	V_{PPL}	V_{PPL}	XXX	XXX

NOTES:
When using the V_{PP} generator, V_{PP1} and V_{PP2} are "don't care."

3.3 Component Management Registers

The Component Management Registers (CMRs) are classified into two categories: those defined by PCMCIA Rev 2.0, and those included by Intel to enhance the interface between the host system and the card's flash memory array. The CMRs provide five control functions: Ready-Busy Mode selection, Voltage Control, Software Write Protection, Card Status and Soft Reset. For more details about the Component Management Register functionality, consult Intel's ***Series 2+ Flash Memory Card User's Manual*** (297373).

3.4 SmartPower

The SmartPower circuitry generates and monitors the card's programming voltages. When a host system does not provide a valid V_{PP} supply, the card's integrated generator can be switched on via the Voltage Control Register. The SmartPower circuitry also detects the host system's V_{CC} level (3.3V or 5.0V) and configures the card's flash memory devices, accordingly driving the 3/5# pin to the memory array to the appropriate value.

The SmartPower circuitry is enabled by writing a "1" to Bit 0 of the Voltage Control Register.

Table 4. Configuration Option Register—PCMCIA (Soft Reset Register)

Attribute Memory Plane Address: 4000H
Read/Write

SRESET	LevIREQ	Configuration Index					
7	6	5	4	3	2	1	0

Default: 02H

Bit 7 = Soft Reset
1 = Reset State
0 = End Reset Cycle

Bit 6 = Level Request
Driven Low

Bits 5-0 = Configuration Index
May Be Written with Values 1-4, Refer to Index in CIS Card Configuration Table Tuple.

Table 5. Card Configuration and Status Register—PCMCIA (Global Power-Down Register)

Attribute Memory Plane Address: 4002H
Read/Write

Reserved					PWRDWN	Reserved	
7	6	5	4	3	2	1	0

Default: 00H

Bit 2 = Power-Down
1 = 28F016SAs in Reset Power-Down
0 = Power-Up

Table 6. Card Status Register—Intel

Attribute Memory Plane Address: 4100H
Read Only

Reserved		SRESET	CMWP	PWRDWN	CISWP	WP	RDY/BSY#
7	6	5	4	3	2	1	0

Default: 01H or 03H

Bit 5 = Soft Reset
1 = Reset State

Bit 4 = Common Memory Write Protect
1 = Write Protected

Bit 3 = Power-Down
1 = Power-Down

Bit 2 = Common Memory CIS Write Protect
1 = Write Protected

Bit 1 = Write Protect Switch
1 = Write Protected

Bit 0 = Card Ready/Busy#
1 = Ready

Table 7. Write Protection Register—Intel

Attribute Memory Plane Address: 4104H
Read/Write

Reserved					BLKEN	CMWP	CISWP
7	6	5	4	3	2	1	0

Default: 04H

Bit 2 = Block Locking Enable
- 1 = Enable Independent 28F016SA Block Locking
- 0 = All Blocks Unlocked

Bit 1 = Common Memory Write Protect
- 1 = Common Memory Minus the CMCIS in Write Protect Status
- 0 = Write Protect According to Independent 28F016SA Block Locking

Bit 0 = Common Memory CIS Write Protect
- 1 = Common Memory CIS in Write Protect Status
- 0 = Write Protect According to Independent 28F016SA Block Locking

Table 8. Voltage Control Register—Intel

Attribute Memory Plane Address: 410CH
Read/Write

V_{CC} LEVEL	Reserved					V_{PP} VALID	V_{PP} GEN
7	6	5	4	3	2	1	0

Default: 82 or 02H

Bit 7 = V_{CC} Level: Read Only Bit
- 1 = Host Supplying 3.3V
- 0 = Host Supplying 5V

Bit 1 = V_{PP} Valid
- 1 = V_{PP} between 11.4V and 12V
- 0 = V_{PP} Invalid.

Bit 0 = V_{PP} Generation
- 1 = Turn on Integrated V_{PP} Generator
- 0 = Turn off Integrated V_{PP} Generator

Note: The V_{PP} Valid bit only reflects the state of the V_{PP} Generator, NOT the external V_{PP}.

Table 9. Ready/Busy Mode Register—Intel

Attribute Memory Plane Address: 4140H
Read/Write

Reserved						RACK	MODE
7	6	5	4	3	2	1	0

Default: 00H

Bit 1 = Ready Acknowledge
- 0 = Clear RDY/BSY#

Bit 0 = RDY/BSY# Mode
- 1 = High-Performance Mode
- 0 = PCMCIA Level Mode

4.0 DEVICE COMMAND SET

The 28F016SA/DD28F032SA-based Series 2+ Command Set increases functionality over earlier 28F008SA-based designs while maintaining backwards compatibility. The extended command set incorporates many new features to improve programmability and write performance such as: page buffered writing, individual block locking, multiple RDY/BSY# configurations and device level queuing capabilities. The following pages list the Series 2+ command set and Bus Cycle Operations overview.

Series 2+ Command Set

Codes (H)	Series 2 Compatible Mode
00H	Invalid/Reserved
10H	Alternate Data Write
20H	Single Block Erase
40H	Data Write
50H	Clear Status Registers
70H	Read CSR
90H	Read ID Codes
B0H	Erase Suspend
D0H	Confirm/Resume
FFH	Read Flash Array

Codes (H)	Series 2+ Performance Enhancement
0CH	Page Buffer Write to Flash
71H	Read GSR or BSRs
72H	Page Buffer Swap
74H	Single Load to Page Buffer
75H	Read Page Buffer
77H	Lock Block
80H	Abort
96H,01H	RY/BY# Level Mode Enable
96H,02H	RY/BY# Pulse-On-Write
96H,03H	RY/BY# Pulse-On Erase
96H,04H	RY/BY# Disable
97H	Upload Status Bits
99H	Upload Device Information
A7H	Erase All Unlocked Blocks
E0H	Sequential Load to Page Buffer
F0H	Sleep

Table 10. 28F008SA-Compatible Mode Command Bus Definitions

Command	First Bus Cycle				Second Bus Cycle			
	R/W	Addr	Data		R/W	Addr	Data	
			Byte	Word			Byte	Word
Read Array	W	DA	FFH	FFFFH	R	DA	AD	AD
Intelligent Identifier	W	DA	90H	9090H	R	IA	ID	ID
Read CSR(1)	W	DA	70H	7070H	R	DA	CSRD	CSRD
Clear Status Register(2)	W	DA	50H	5050H				
Word/Byte Write §	W	WA	40H	4040H	W	WA	WD	WD
Word/Byte Write (Alternate) §	W	WA	10H	1010H	W	WA	WD	WD
Block Erase/Confirm §	W	BA	20H	2020H	W	BA	D0H	D0D0H
Erase Suspend/Resume	W	DA	B0H	B0B0H	W	DA	D0H	D0D0H

ADDRESSES:
DA Device Address
BA Block Address
IA Identifier Address
WA Write Address

DATA:
AD Array Data
CSRD CSR Data
ID Identifier Data
WD Write Data

§ = Queueable Commands

NOTES:
1. The CSR is automatically available after the device enters Data Write, Erase or Suspend operations.
2. Clears CSR.3, CSR.4 and CSR.5. Also clears GSR.5 and all BSR.5 and BSR.2 bits.

Table 11. 28F016SA-Super-Set Mode Performance Enhancement Command Bus Definitions

Command	Notes	First Bus Cycle				Second Bus Cycle				Third Bus Cycle			
		Oper	Addr	Data		Oper	Addr	Data		Oper	Addr	Data	
				Byte	Word			Byte	Word			Byte	Word
Read Page Buffer		W	DA	75H	7575H	R	PA	PD	PDPD				
Page Buffer Swap	6	W	DA	72H	7272H								
Single Load to Page Buffer		W	DA	74H	7474H	W	PA	PD	PDPD				
Sequential Load to Page Buffer	4,5	W	DA	E0H	E0E0H	W	DA	BCH		W	DA	BCH	
Page Buffer Write to Flash Array §	3,4,5	W	DA	0CH	0C0CH	W	A_0	BC(L,H)		W	WA	BC(H,L)	
RY/BY# Pulse-On-Erase §	7	W	DA	96H	9696H	W	DA	D0H	D0D0H				
RY/BY# Pulse-On-Write §	7	W	DA	96H	9696H	W	DA	D1H	D1D1H				
RY/BY# Enable to Level-Mode §	7	W	DA	96H	9696H	W	DA	D2H	D2D2H				
RY/BY# Disable §	7	W	DA	96H	9696H	W	DA	D3H	D3D3H				
Lock Block/ Confirm §		W	DA	77H	7777H	W	BA	D0H	D0D0H				
Upload Status Bits/ Confirm §	2	W	DA	97H	9797H	W	DA	D0H	D0D0H				
Read Extended Status Registers	1	W	DA	71H	7171H	R	RA	GSRD/BSRD					
Erase All Unlocked Blocks/ Confirm §		W	DA	A7H	A7A7H	W	DA	D0H	D0D0H				
Sleep		W	DA	F0H	F0F0H								
Abort		W	DA	80H	8080H								
Upload Device Information		W	DA	99H	9999H	W	DA	D0H	D0D0H				

ADDRESSES:
DA Device Address
BA Block Address
IA Identifier Address
PA Page Buffer Address
RA Extended Register Address
WA Write Address
X Don't Care

DATA:
AD Write Address
CSRD CSR Data
G/BSRD GSR/BSR Data
ID Identifier Data
WD Write Data
PD Page Buffer Data

DATA COUNTS:
WC(L,H) Word Count (Low, High)
BC(L,H) Byte Count (Low, High)
WD(L,H)V Write Data (Low, High)

§ = Queueable Commands

NOTES:
1. RA can be the GSR address or any BSR address.
2. Upon device power-up, all BSR lock-bits are locked. The Lock Status Upload command must be written to reflect the actual lock-bit status.
3. A_0 is automatically complemented to load the second byte of data.
4. BCH/WCH must be at 00H for this product because of the 256-byte Page Buffer size *and* to avoid writing the Page Buffer contents into more than one 256-byte segment within an array block. They are simply shown for Page Buffer expandability.
5. PA and PD (whose count is given in cycles 2 and 3) are supplied starting in the fourth cycle (not shown).
6. This command allows the user to swap between available page buffers (0 or 1).
7. These commands reconfigure RY/BY# output to one of two pulse modes, or they enable and disable the RY/BY# function.

5.0 DEVICE STATUS REGISTER

Each 28F016SA has three types of status registers: the Compatible Status Register (CSR), the Global Status Register (GSR) and the Block Status Register (BSR). The Compatible Status Register is identical to the 28F008SA Status Register. The Global Status Register contains queue and page buffer information about each device. Each block within the device has a Block Status Register assigned to it. The Block Status Register contains the Block Locking Status and other information specific to the block being addressed.

Table 12. Compatible Status Register

Read Only Register

WSMS	ESS	ES	DWS	VPPS	Reserved		
7	6	5	4	3	2	1	0

Default: 80H

CSR.7 = WRITE STATE MACHINE STATUS (WSMS)
1 = Ready
0 = Busy

CSR.6 = ERASE-SUSPEND STATUS (ESS)
1 = Erase Suspended
0 = Erase in Progress/Completed

CSR.5 = ERASE STATUS (ES)
1 = Error in Block Erasure
0 = Successful Block Erase

CSR.4 = DATA-WRITE STATUS (DWS)
1 = Error in Data Write
0 = Data Write Successful

CSR.3 = V_{PP} STATUS (VPPS)
1 = V_{PP} Low Detect, Operation Abort
0 = V_{PP} OK

6

Table 13. Global Status Register

WSMS	ESS	ES	DWS	VPPS	Reserved		
7	6	5	4	3	2	1	0

Default: 80H

CSR.7 = WRITE STATE MACHINE STATUS (WSMS)
- 1 = Ready
- 0 = Busy

CSR.6 = ERASE-SUSPEND STATUS (ESS)
- 1 = Erase Suspended
- 0 = Erase in Progress/Completed

CSR.5 = ERASE STATUS (ES)
- 1 = Error in Block Erasure
- 0 = Successful Block Erase

CSR.4 = DATA-WRITE STATUS (DWS)
- 1 = Error in Data Write
- 0 = Data Write Successful

CSR.3 = V_{PP} STATUS (VPPS)
- 1 = V_{PP} Low Detect, Operation Abort
- 0 = V_{PP} OK

Table 14. Block Status Register

BS	BLS	BOS	BOAS	QS	VPPS	Reserved	
7	6	5	4	3	2	1	0

Default: 80H

BSR.7 = BLOCK STATUS (BS)
- 1 = Ready
- 0 = Busy

BSR.6 = BLOCK-LOCK STATUS (BLS)
- 1 = Block Unlocked for Write/Erase
- 0 = Block Locked for Write/Erase

BSR.5 = BLOCK OPERATION STATUS (BOS)
- 1 = Operation Unsuccessful
- 0 = Operation Successful or Currently Running

BSR.4 = BLOCK OPERATION ABORT STATUS (BOAS)
- 1 = Operation Aborted
- 0 = Operation Not Aborted

BSR.3 = QUEUE STATUS (QS)
- 1 = Queue Full
- 0 = Queue Available

BSR.2 = V_{PP} STATUS (VPPS)
- 1 = V_{PP} Low Detect, Operation Abort
- 0 = V_{PP} OK

6.0 PCMCIA CARD INFORMATION STRUCTURE

The Card Information Structure (CIS) begins at address 00000000H of the card's Attribute Memory Plane. It contains a variable length chain of data blocks (tuples) that conform to a basic format (Table 15). The CIS of the Series 2+ Flash Memory Card is found in Table 16.

Table 15. PCMCIA Tuple Format

Bytes	Data
0	Tuple Code: CISTPL_xxx. The tuple code 0FFH indicates no more tuples in the list.
1	Tuple Link: TPL_LINK. Link to the next tuple in the list. This can be viewed as the number of additional bytes in tuple, excluding this byte. A link field of zero indicates an empty tuple body. A link field containing 0FFH indicates the last tuple in the list.
2-n	Bytes specific to this tuple.

Table 16. Tuples for Series 2+ Card

Address	Value	Description
00h	01H	**CISTPL_DEVICE**
02h	04H	TPL_LINK
04h	57H	FLASH
06h	22H 32H	150 ns 250 nx
08h	 0EH 1EH 4EH 9EH	CARD SIZE 4 MB 8 MB 20 MB 40 MB
0AH	FFH	END OF DEVICE
0CH	1CH	**CISTPL_DEVICE_OC**
0EH	05H	TPL_LINK
10H	02H	OTHER CONDITIONS - 3 V_{CC}
12H	57H	FLASH
14H	32H	250 ns
16H	 0EH 1EH 4EH 9EH	CARD SIZE 4 MB 8 MB 20 MB 40 MB
18H	FFH	END OF DEVICE
1AH	17H	**CISTPL_DEVICE_A**
1CH	04H	TPL_LINK
1EH	1FH	ROM
20H	22H	150 ns
22H	01H	2 Kb
24H	FFH	END OF DEVICE
26H	1DH	**CISTPL_DEVICE_OA**
28H	05H	TPL_LINK
2AH	02H	OTHER CONDITIONS - 3 V_{CC}
2CH	17H	ROM
2EH	32H	250 ns
30H	01H	2 Kb
32H	FFH	END OF DEVICE
34H	18H	**CISTPL JEDEC_C**
36H	02H	TPL_LINK
38H	89H	INTEL J-ID

Address	Value	Description
3AH	A0H	28F016 J-ID
3CH	00H	NULL CONTROL TUPLE
3EH	15H	**CISTPL VERS 1**
40H	39H	TPL_LINK
42H	04H	TPLLV1_MAJOR
44H	01H	TPLLV1_MINOR
46H	49H	TPLLV1_INFO I
48H	6EH	n
4AH	74H	t
4CH	65H	e
4EH	6CH	l
50H	00H	END TEXT
52H	53H	S
54H	32H	2
56H	45H	E
58H	34H 38H 32H 34H	4 MB 8 MB 20 MB 40 MB
5AH	20H 20H 30H 30H	4 MB 8 MB 20 MB 40 MB
5CH	53H	S
	20H	SPACE
5EH	57H	W
	20H	SPACE
60H	00H	ENDTEXT
62H	43H	C
64H	4FH	O
66H	50H	P
68H	59H	Y
6AH	52H	R
6CH	49H	I
6EH	47H	G
70H	48H	H
72H	54H	T

Table 16. Tuples for Series 2+ Card (Continued)

Address	Value	Description
74H	20H	SPACE
76H	49H	I
78H	6EH	n
7AH	74H	t
7CH	65H	e
7EH	6CH	l
80H	20H	SPACE
82H	43H	C
84H	4FH	O
86H	52H	R
88H	50H	P
8AH	4FH	O
8CH	52H	R
8EH	41H	A
90H	54H	T
92H	49H	I
94H	4FH	O
96H	4EH	N
98H	20H	SPACE
9AH	31H	1
9CH	39H	9
9EH	39H	9
A0H	33H	3
A2H	20H	SPACE
A4H	47H	G
A6H	4CH	L
A8H	41H	A
AAH	44H	D
ACH	45H	E
AEH	4BH	K
B0H	00H	END TEXT
B2H	FFH	END OF LIST
B4H	1AH	**CISTPL_CONF**
B6H	05H	TUPL_LINK
B8H	01H	TPCC_SZ
BAH	04H	TPCC_LAST

Address	Value	Description
BCH	00H	TPCC_RADR
BEH	40H	TPCC_RADR
C0H	03H	TPCC_RMSK
C2H	00H	NULL CONTROL TUPLE
C4H	1BH	**CISTPL_CFTABLE_ENTRY**
C6H	08H	TPL_LINK
C8H	01H	**TPCE_INDEX (01H)**
CAH	01H	TPCE_FS (V_{CC} ONLY)
CCH	79H	TPCE_PD V_{CC} PARAMETER SELECTION BYTE
CEH	55H	V_{CC} NOMINAL VOLTAGE 5V ± 5%
D0H	53H	I_{CC} STATIC 500 µA
D2H	1EH	I_{CC} AVERAGE 150 mA
D4H	1EH	I_{CC} PEAK 150 mA
D6H	1BH	I_{CC} PWRDWN 200 µA
D8H	1BH	**CISTPL_CFTABLE_ENTRY**
DAH	0FH	TPL_LINK
DCH	02H	**TPCE_INDEX (02H)**
DEH	02H	TPCE_FS (V_{CC} AND V_{PP})
E0H	79H	TPCE_PD V_{CC} PARAMETER SELECTION BYTE
E2H	55H	V_{CC} NOMINAL VOLTAGE 5V ± 5%
E4H	2BH	I_{CC} STATIC 250 µA
E6H	06H	I_{CC} AVERAGE 100 mA
E8H	06H	I_{CC} PEAK 100 mA
EAH	52H	I_{CC} PWRDWN 50 µA
		TPCE_PD
ECH	79H	V_{PP} PARAMETER
		SELECTION BYTE
EEH	8EH	12.0V ± 5%
F0H	7DH	NC OK ON STANDBY & PWD
F2H	53H	I_{PP} STATIC 500 µA
F4H	25H	I_{PP} AVERAGE 20 mA
F6H	25H	I_{PP} PEAK 20 mA

Table 16. Tuples for Series 2+ Card (Continued)

Address	Value	Description
F8H	52H	I_{PP} PWRDWN 50 μA
FAH	1BH	**CISTPL_CFTABLE_ENTRY**
FCH	09H	TPL_LINK
FEH	03H	**TPCE_INDEX (03H)**
100H	01H	TPCE_FS (V_{CC} ONLY) TPCE_PD
102H	79H	V_{CC} PARAMETER SELECTION BYTE
104H	B5H	V_{CC} = 3.3V
106H	1EH	EXTENSION BYTE
108H	04H	I_{CC} STATIC 1 mA
10AH	1EH	I_{CC} AVERAGE 150 mA
10CH	1EH	I_{CC} PEAK 150 mA
10EH	53H	I_{CC} PWRDWN 500 μA
110H	1BH	**CISTPL_CFTABLE_ENTRY**
112H	10H	TPL_LINK
114H	04H	**TPCE_INDEX (04H)**
116H	02H	TPCE_FS (V_{CC} AND V_{PP})
118H	79H	TPCE_PD V_{CC} PARAMETER SELECTION BYTE
11AH	B5H	V_{CC} = 3.3V
11CH	1EH	EXTENSION BYTE
11EH	2BH	I_{CC} STATIC 250 μA
120H	06H	I_{CC} AVERAGE 100 mA
122H	06H	I_{CC} PEAK 100 mA
124H	52H	I_{CC} PWRDWN 50 μA
126H	79H	TPCE_PD V_{PP} PARAMETER SELECTION BYTE
128H	8EH	12.0V = /- 5%
12AH	7DH	NC OK ON STANDBY & PWD
12CH	53H	I_{PP} STATIC 500 μA
12EH	25H	I_{PP} AVERAGE 20 mA
130H	25H	I_{PP} PEAK 20 mA
132H	1BH	I_{PP} PWRDWN 150 μA
134H	00H	NULL CONTROL TUPLE

Address	Value	Description
136H	00H	NULL CONTROL TUPLE
138H	1EH	**CISTPL DEVICEGEO**
13AH	06H	TPL_LINK
13CH	02H	DGTPL_BUS
13EH	11H	DGTPL_EBS
140H	01H	DGTPL_RBS
142H	01H	DGTPL_WBS
144H	01H	DGTPL_PART = 1
146H	01H	FLASH DEVICE INTERLEAVE
148H	20H	**CISTPL_MANFID**
14AH	04H	TPL_LINK (04H)
		TPLMID_MANF
14CH	89H	LSB
14EH	00H	MSB
150H	12H 22H 42H 62H	4 MB - 150 ns 8 MB - 150 ns 20 MB - 150 ns 40 MB - 150 ns
152H	84H	TPLMID_CARD MSB
154H	21H	**CISTPL_FUNCID**
156H	02H	TPL_LINK
158H	01H	TPLFID_FUNCTION (MEMORY)
15AH	00H	TPLFID_SYSINIT (NONE)
15CH	FFH	**CISTPL_END**
	00H	INVALID ADDRESS
		(156H-1FEH)

6

7.0 SYSTEM DESIGN CONSIDERATIONS

7.1 Power Supply Decoupling

Flash memory power-switching characteristics require careful device decoupling. System designers are interested in three supply current issues: standby, active and transient current peaks which are produced by rising and falling edges of CE_1# and CE_2#. The capacitive and inductive loads on the card and internal flash memory device pairs determine the magnitudes of these peaks.

Three-line control and proper decoupling capacitor selection suppress transient voltage peaks. Series 2+ cards contain on-card ceramic decoupling capacitors connected between V_{CC} and GND, and between V_{PP1}/V_{PP2} and GND.

The card connector should also have a 4.7 µF electrolytic capacitor between V_{CC} and GND, as well as between V_{PP1}/V_{PP2} and GND. The bulk capacitors overcome voltage slumps caused by printed-circuit-board trace inductance, and supply charge to the smaller capacitors as needed.

7.2 Power-Up/Down Protection

The PCMCIA/JEIDA-specified socket properly sequences the power supplies to the flash memory card via shorter and longer pins. This design assures that hot insertion and removal will not result in card damage or data loss.

Each device in the memory card is designed to offer protection against accidental erasure or writing, caused by spurious system-level signals that may exist during power transitions. The card will power-up into the read state.

A system designer must guard against active writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and CE_1# must be low for a command write, driving either to V_{IH} will inhibit writes. With its control register architecture, alteration of device contents only occurs after successful completion of the two-step command sequences. While these precautions are sufficient for most applications, an alternative approach would allow V_{CC} to reach its steady state value before raising V_{PP1}/V_{PP2} above V_{CC} + 2.0V. In addition, upon powering down, V_{PP1}/V_{PP2} should be below V_{CC} + 2.0V before lowering V_{CC}.

NOTE:
The Integrated V_{PP} generator defaults to the power off condition after reset and system power-up. The Vpp Generation circuitry must be enabled for the memory card to operate in 3.3V-Only or 5.0V-Only mode.

7.3 Hot Insertion/Removal

The capability to remove or insert PC cards while the system is powered on (i.e., hot insertion/removal) requires careful design approaches on the system and card levels. To design for this capability, consider card over-voltage stress, system power fluxuations and control line stability.

8.0 ELECTRICAL SPECIFICATIONS

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

8.1 Absolute Maximum Ratings*

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

Operating Temperature
During Read .....0°C to +60°C(1)
During Write .....0°C to +60°C

Storage Temperature.....−30°C to +70°C(2)

Voltage on Any Pin with
Respect to Ground .....−2.0V to V_{CC} + 2.0V(2)

V_{PP1}/V_{PP2} Supply Voltage with
Respect to Ground ..−2.0V to V_{CC} + 14.0V(2, 3)

V_{CC} Supply Voltage with
Respect to Ground .....−0.5V to +7.0V

NOTES:
1. Operating temperature is for commercial product defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to 2.0V for periods less than 20 ns. Maximum DC voltage on output pins is V_{CC} + 0.5V, which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.
3. Maximum DC input voltage on V_{PP1}/V_{PP2} may overshoot to +14.0V for periods less than 20 ns.
4. V_{PP} generator turned "on" for 3.3V or 5.0V only operation.

Symbol	Parameter	Min	Max	Units
V_{CC} at 3.3V, 12V V_{PP}	V_{CC} Supply Voltage (±0.3V)	3.0	3.6	V
V_{CC} at 3.3V, V_{PP} Gen(4)	V_{CC} Supply Voltage (±0.15V)	3.15	3.45	V
V_{CC} at 5.0V	V_{CC} Supply Voltage (±0.25V)	4.75	5.25	V

6

9.0 SERIES 2+ DC CHARACTERISTICS

9.1 General

Symbol	Parameter	Notes	Min	Max	Units	Test Conditions
I_{LI}	Input Leakage Current	1, 3		±20	μA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or GND
I_{LO}	Output Leakage Current	1		±20	μA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or GND
V_{IL5}	Input Low Voltage	1	0	0.8	V	V_{CC}= 5V
$V_{IL3.3}$				0.7		V_{CC} = 3.3V
V_{IH5}	Input High Voltage	1	2.4	V_{CC} + 0.5	V	V_{CC} = 5V
$V_{IH3.3}$			2.2	V_{CC} + 0.3		V_{CC} = 3.3V
V_{OL}	Output Low Voltage	1		0.4	V	I_{OL} = 3.2 mA
V_{OH}	Output High Voltage	1	V_{CC} − 0.4	V_{CC}	V	I_{OH} = −2.0 mA
V_{PPL}	V_{PP} during Read Only Operations	1, 2	0	6.5	V	
V_{PPH}	V_{PP} during Read/Write Operations	1	11.4	12.6	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage	1	2.0		V	

NOTES:
1.Values are the same for byte and word wide modes for all card densities.
2.Block Erases/Data Writes are inhibited when V_{PP} and V_{PPL} are not guaranteed in the range between V_{PPH} and V_{PPL}.
3.Exceptions: With V_{IN} = GND, the leakage current on CE_1#, CE_2#, REG#, OE#, and WE# will be < 500 μA due to internal pull-up resistors. With $V_{IN} = V_{CC}$, RST leakage current will be < 500 μA due to internal pull-down resistors. With $V_{IN} = V_{CC}$, A_{21}–A_{25} leakage current will be <100 μA due to internal pull down resistors.

9.2 CMOS Interfacing: V_{CC} = 3.3V

Series 2+ DC Characteristics(1)

Symbol	Parameter	Density (Mbytes)	Notes	x8 Mode		x16 Mode		Units	Test Conditions
				Typ	Max	Typ	Max		
I_{CCR}	V_{CC} Read Current	4, 8, 20, 40	2, 3		75		100	mA	$V_{CC} = V_{CC}$ Max t_{CYCLE} = 200 ns
I_{CCW}	V_{CC} Write Current	4, 8, 20, 40	2, 3, 4		40		50	mA	V_{PP} Gen = OFF during Data Write
			2, 3, 5		100		175	mA	V_{PP} Gen = ON during Data Write
I_{CCE}	V_{CC} Erase Current	4, 8, 20, 40	2, 3, 4		40		50	mA	V_{PP} Gen = OFF
			2, 3, 5		80		150	mA	V_{PP} Gen = ON
I_{CCSL}	V_{CC} Sleep Current	4	3, 4, 6	12	20	12	20	µA	$V_{CC} = V_{CC}$ Max Control Signals = V_{CC}
		8		14	30	14	30		
		20		20	60	20	60		
		40		30	110	30	110		
I_{CCS}	V_{CC} Standby Current	4	2, 3, 4, 6	75	115	110	210	µA	$V_{CC} = V_{CC}$ Max Control Signals = V_{CC}
		8		80	125	115	230		
		20		85	155	120	250		
		40		100	200	150	300		
I_{PPW}	V_{PP} Write Current	4, 8, 20, 40	2, 3, 4	10	15	20	30	mA	Data Write in Progress $V_{PP} = V_{PPH}$
I_{PPE}	V_{PP} Erase Current	4, 8, 20, 40	2, 3, 4	6	12	12	22	mA	Block (Pair) Erase in Progress $V_{PP} = V_{PPH}$
I_{PPSL}	V_{PP} Sleep Current	4, 8, 20, 40	2, 3, 4	0	0.5	0	0.5	mA	$V_{PP} \leq V_{CC}$
I_{PPS}	V_{PP} Standby/ Read Current	4, 8, 20, 40	2, 3, 4	0	0.5	0	0.5	mA	$V_{PP} \leq V_{CC}$

CMOS Test Conditions: V_{IL} = GND ± 0.2V V_{IH} = V_{CC} ± 0.2V

NOTES:
1. All currents are RMS values unless otherwise specified. Typical V_{CC} = 5V, V_{PP} = 12V, T = 25°C.
2. Two devices active in word mode, one device active in byte mode.
3. Devices not addressed are in sleep mode.
4. V_{PP} Generation Circuitry turned off.
5. V_{PP} Generation Circuitry turned on.
6. Control Signals, CE_1#, CE_2#, OE#, WE#, REG#.

9.3 CMOS Interfacing: V_{CC} = 5.0V

Series 2+ DC Characteristics(1)

Symbol	Parameter	Density (Mbytes)	Notes	x8 Mode		x16 Mode		Units	Test Conditions
				Typ	Max	Typ	Max		
I_{CCR}	V_{CC} Read Current	4, 8, 20, 40	2, 3		140		160	mA	$V_{CC} = V_{CC}$ Max t_{CYCLE} = 200 ns
I_{CCW}	V_{CC} Write Current	4, 8, 20, 40	2, 3, 4		85		120	mA	V_{PP} Gen = OFF during Data Write
			2, 3, 5		120		150	mA	V_{PP} Gen = ON during Data Write
I_{CCE}	V_{CC} Erase Current	4, 8, 20, 40	2, 3, 4		75		100	mA	V_{PP} Gen = OFF
			2, 3, 5		100		150	mA	V_{PP} Gen = ON
I_{CCSL}	V_{CC} Sleep Current	4	3, 4, 6	12	20	12	20	µA	$V_{CC} = V_{CC}$ Max Control Signals = V_{CC}
		8		14	30	14	30		
		20		20	60	20	60		
		40		30	110	30	110		
I_{CCS}	V_{CC} Standby Current	4	2, 3, 4, 6	75	115	110	210	µA	$V_{CC} = V_{CC}$ Max Control Signals = V_{CC}
		8		80	125	115	230		
		20		85	155	120	250		
		40		100	200	150	300		
I_{PPW}	V_{PP} Write Current ($V_{PP} = V_{PPH}$)	4, 8, 20, 40	2, 3, 4	6	12	14	24	mA	Data Write in Progress $V_{PP} = V_{PPH}$
I_{PPE}	V_{PP} Erase Current	4, 8, 20, 40	2, 3, 4	6	12	12	22	mA	Block (Pair) Erase in Progress $V_{PP} = V_{PPH}$
I_{PPSL}	V_{PP} Sleep Current	4, 8, 20, 40	2, 3, 4	0	0.5	0	0.5	mA	$V_{PP} \leq V_{CC}$
I_{PPS}	V_{PP} Standby/ Read Current	4, 8, 20, 40	2, 3, 4	0	0.5	0	0.5	mA	$V_{PP} \leq V_{CC}$

CMOS Test Conditions: V_{IL} = GND ± 0.2V, V_{IH} = V_{CC} ± 0.2V

NOTES:
1. All currents are RMS values unless otherwise specified. Typical V_{CC} = 5V, V_{PP} = 12V, T = 25°C.
2. Two devices active in word mode, one device active in byte mode.
3. Devices not addressed are in sleep mode.
4. V_{PP} Generation Circuitry turned off.
5. V_{PP} Generation Circuitry turned on.
6. Control Signals, CE_1#, CE_2#, OE#, WE#, REG#.

9.4 TTL Interfacing: V_{CC} = 3.3V

Series 2+ DC Characteristics(1)

Symbol	Parameter	Density (Mbytes)	Notes	x8 Mode		x16 Mode		Units	Test Conditions
				Typ	Max	Typ	Max		
I_{CCR}	V_{CC} Read Current	4, 8, 20, 40	2, 3		75		90	mA	$V_{CC} = V_{CC}$ Max t_{CYCLE} = 200 ns
I_{CCW}	V_{CC} Write Current	4, 8, 20, 40	2, 3, 4		85		100	mA	V_{PP} Gen = OFF during Data Write
			2, 3, 5		150		225	mA	V_{PP} Gen = ON during Data Write
I_{CCE}	V_{CC} Erase Current	4, 8, 20, 40	2, 3, 4		85		100	mA	V_{PP} Gen = OFF
			2, 3, 5		125		180	mA	V_{PP} Gen = ON
I_{CCSL}	V_{CC} Sleep Current	4, 8, 20, 40	3, 4, 6		70		70	mA	$V_{CC} = V_{CC}$ Max Control Signals = V_{IH}
I_{CCS}	V_{CC} Standby Current	4, 8, 20, 40	2, 3, 4, 6		70		70	mA	$V_{CC} = V_{CC}$ Max Control Signals = V_{IH}
I_{PPW}	V_{PP} Write Current	4, 8, 20, 40	2, 3, 4	10	15	20	30	mA	Data Write in Progress $V_{PP} = V_{PPH}$
I_{PPE}	V_{PP} Erase Current	4, 8, 20, 40	2, 3, 4	5	10	10	20	mA	Block (Pair) Erase in Progress $V_{PP} = V_{PPH}$
I_{PPSL}	V_{PP} Sleep Current	4, 8, 20, 40	2, 3, 4	0	0.5	0	0.5	mA	$V_{PP} \leq V_{CC}$
I_{PPS}	V_{PP} Standby/ Read Current	4, 8, 20, 40	2, 3, 4	0	0.5	0	0.5	mA	$V_{PP} \leq V_{CC}$

TTL Test Conditions: V_{IL} = 0.7V, V_{IH} = 2.2V

NOTES:
1. All currents are RMS values unless otherwise specified. Typical V_{CC} = 5V, V_{PP} = 12V, T = 25°C.
2. Two devices active in word mode, one device active in byte mode.
3. Devices not addressed are in sleep mode.
4. V_{PP} Generation Circuitry turned off.
5. V_{PP} Generation Circuitry turned on.
6. Control Signals, CE_1#, CE_2#, OE#, WE#, REG#.

9.5 TTL Interfacing: V_{CC} = 5.0V

Series 2+ DC Characteristics(1)

Symbol	Parameter	Density (Mbytes)	Notes	x8 Mode		x16 Mode		Units	Test Conditions
				Typ	Max	Typ	Max		
I_{CCR}	V_{CC} Read Current	4, 8, 20, 40	2, 3		170		190	mA	V_{CC} = V_{CC} Max t_{CYCLE} = 150 ns
I_{CCW}	V_{CC} Write Current	4, 8, 20, 40	2, 3, 4		135		170	mA	V_{PP} Gen = OFF during Data Write
			2, 3, 5		170		250	mA	V_{PP} Gen = ON during Data Write
I_{CCE}	V_{CC} Erase Current	4, 8, 20, 40	2, 3, 4		125		150	mA	V_{PP} Gen = OFF
			2, 3, 5		150		200	mA	V_{PP} Gen = ON
I_{CCSL}	V_{CC} Sleep Current	4, 8, 20, 40	3, 4, 6		100		100	mA	V_{CC} = V_{CC} Max Control Signals = V_{IH}
I_{CCS}	V_{CC} Standby Current	4, 8, 20, 40	2, 3, 4, 6		100		100	mA	V_{CC} = V_{CC} Max Control Signals = V_{IH}
I_{PPW}	V_{PP} Write Current	4, 8, 20, 40	2, 3, 4	7	12	14	24	mA	Data Write in Progress V_{PP} = V_{PPH}
I_{PPE}	V_{PP} Erase Current	4, 8, 20, 40	2, 3, 4	5	10	10	20	mA	Block (Pair) Erase in Progress V_{PP} = V_{PPH}
I_{PPSL}	V_{PP} Sleep Current	4, 8, 20, 40	2, 3, 4	0	0.5	0	0.5	mA	$V_{PP} \leq V_{CC}$
I_{PPS}	V_{PP} Standby/ Read Current	4, 8, 20, 40	2, 3, 4	0	0.5	0	0.5	mA	$V_{PP} \leq V_{CC}$

TTL Test Conditions: V_{IL} = 0.8V, V_{IH} = 2.4V

NOTES:
1. All currents are RMS values unless otherwise specified. Typical V_{CC} = 5V, V_{PP} = 12V, T = 25°C.
2. Two devices active in word mode, one device active in byte mode.
3. Devices not addressed are in sleep mode.
4. V_{PP} Generation Circuitry turned off.
5. V_{PP} Generation Circuitry turned on.
6. Control Signals, CE_1#, CE_2#, OE#, WE#, REG#.

10.0 AC CHARACTERISTICS

AC Timing Diagrams and characteristics are guaranteed to meet or exceed PCMCIA 2.1 specifications. No delay occurs when switching between the Common and Attribute Memory Planes.

10.1 Read Operations: Common and Attribute Memory

Symbol		Parameter	150 ns at 5V		250 ns at 3.3V		Unit
JEDEC	PCMCIA		Min	Max	Min	Max	
t_{AVAV}	t_{RC}	Read Cycle Time	150		250		ns
t_{AVQV}	t_a (A)	Address Access Time		150		250	ns
t_{ELQV}	t_a (CE)	Card Enable Access Time		150		250	ns
t_{GLQV}	t_a (OE)	Output Enable Access Time		75		125	ns
t_{EHQX}	t_{dis} (CE)	Output Disable Time from CE#		75		100	ns
t_{GHQZ}	t_{dis} (OE)	Output Disable Time from OE#		75		100	ns
t_{GLQX}	t_{en} (CE)	Output Enable Time from CE#	5		5		ns
t_{ELQX}	t_{en} (OE)	Output Enable Time from OE#	5		5		ns
t_{PHQV}		Power-Down Recovery to Output Delay. V_{CC} = 5V		530		670	ns

6

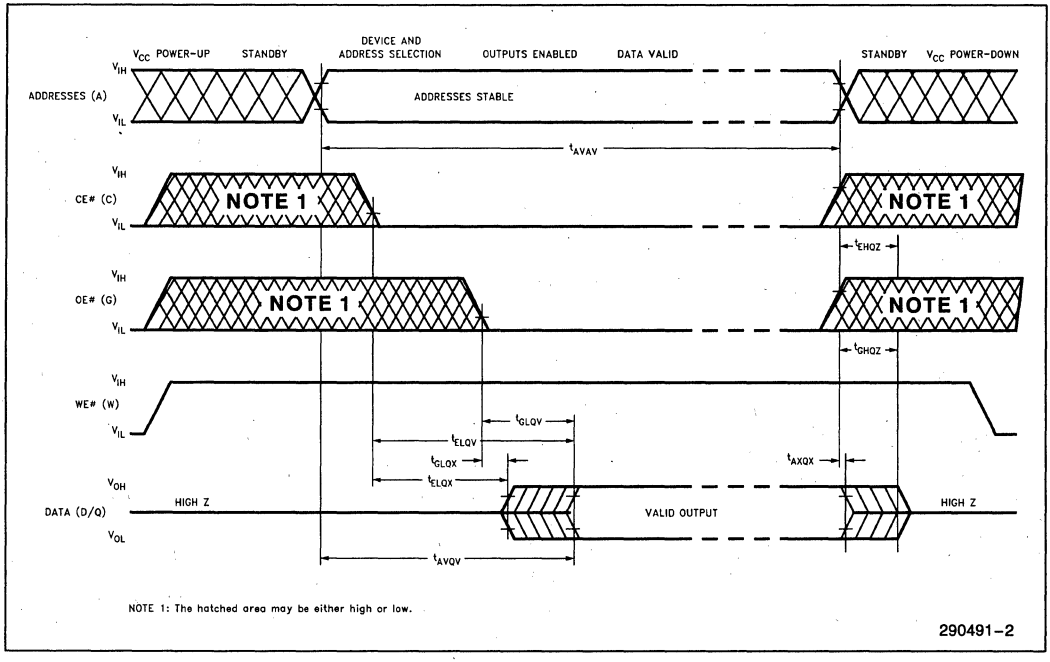


Figure 3. AC Waveforms for Read Operations

10.2 Write Operations: Common and Attribute Memory (1)

Symbol		Parameter	150 ns at 5V		250 ns at 3.3V		Unit
JEDEC	PCMCIA		Min	Max	Min	Max	
t_{AVAV}	t_cW	Write Cycle Time	150		250		ns
t_{WLWH}	t_w (WE)	Write Pulse Width	80		150		ns
t_{AVWL}	t_{su} (A)	Address Setup Time	20		30		ns
t_{AVWH}	t_{su} (A-WEH)	Address Setup Time for WE#	100		180		ns
t_{VPWH}	t_{vps}	V_{PP} Setup to WE# Going High	100		180		ns
t_{ELWH}	t_{su} (CEWEH)	Card Enable Setup Time for WE#	100		180		ns
t_{DVWH}	t_{su} (D-WEH)	Data Setup Time for WE#	50		80		ns
t_{WHDX}	t_h (D)	Data Hold Time	20		30		ns
t_{WHAX}	t_{rec} (WE)	Write Recover Time	20		30		ns
t_{WHRL}		WE# High to RDY/BSY#		140		140	ns
t_{QVVL}		V_{PP} Hold from Operation Complete	0		0		ns
t_{WHGL}	t_h (OE-WE)	Output Enable Hold from WE#	80		120		ns
t_{PHWL}		Power-Down Recovery to WE# Going Low		1		1	µs

NOTE:
1. Read timing characteristics during Erase and Data Write operations are the same as during Read-Only operations. Refer to AC Characteristics for Read-Only operations.

6

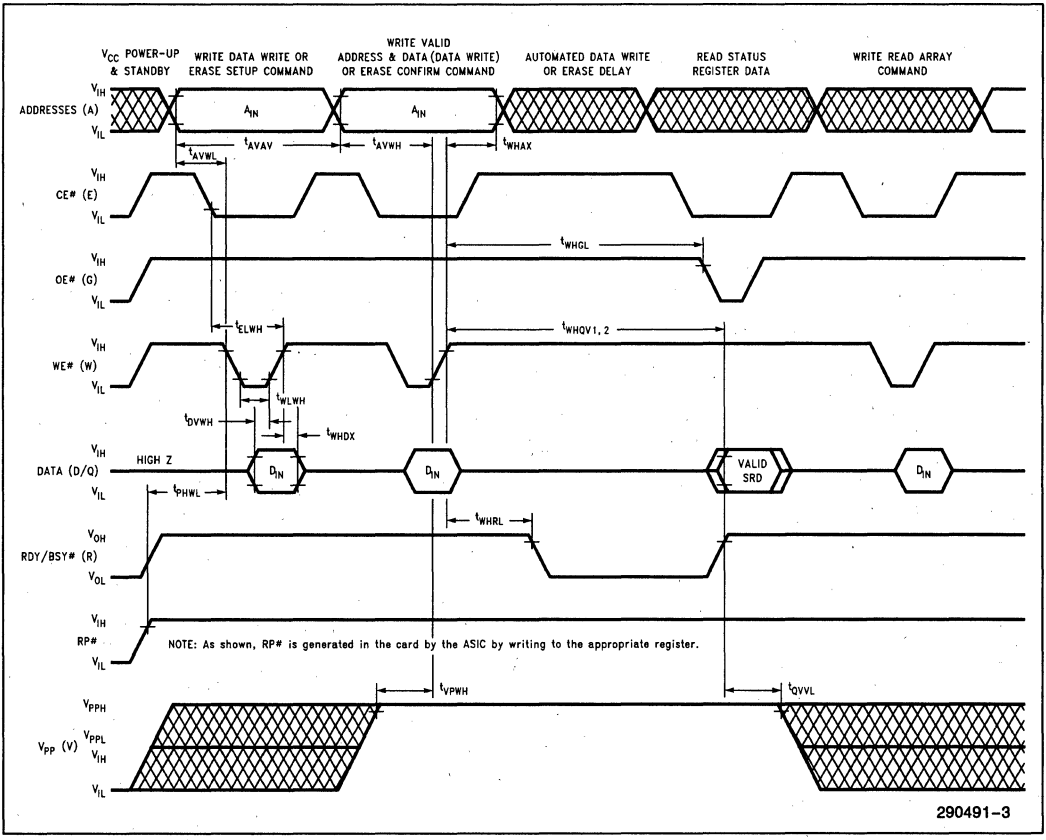


Figure 4. AC Waveforms for Write Operations

10.3 CE#-Controlled Write Operations: Common and Attribute Memory

Symbol		Parameter	150 ns at 5V		250 ns at 3.3V		Unit
JEDEC	PCMCIA		Min	Max	Min	Max	
t_{AVAV}	t_cW	Write Cycle Time	150		250		ns
t_{ELEH}	t_w (WE)	Chip Enable Pulse Width	80		150		ns
t_{AVEL}	t_{su} (A)	Address Setup Time	20		30		ns
t_{AVEH}	t_{su} (A-WEH)	Address Setup Time for CE#	100		180		ns
t_{VPEH}	t_{vps}	V_{PP} Setup to CE# Going High	100		180		ns
t_{WLEH}	t_{su}(CE-WEH)	Write Enable Setup Time for CE#	100		180		ns
t_{DVEH}	t_{su} (D-WEH)	Data Setup Time for CE#	50		60		ns
t_{EHDX}	t_h (D)	Data Hold Time	20		30		ns
t_{EHAX}	t_{rec} (WE)	Write Recover Time	20		30		ns
t_{EHRL}		CE# High to RDY/BSY#		140		140	ns
t_{QVVL}		V_{PP} Hold from Operation Complete	0		0		ns
t_{EHGL}	t_h (OE-WE)	Output Enable Hold from WE#	80		120		ns
t_{PHEL}		Power-Down Recovery to CE# Going Low		1		1	μs

NOTE:
1. Read timing characteristics during Erase and Data Write operations are the same as during Read-Only operations. Refer to AC Characteristics for Read-Only operations.

6

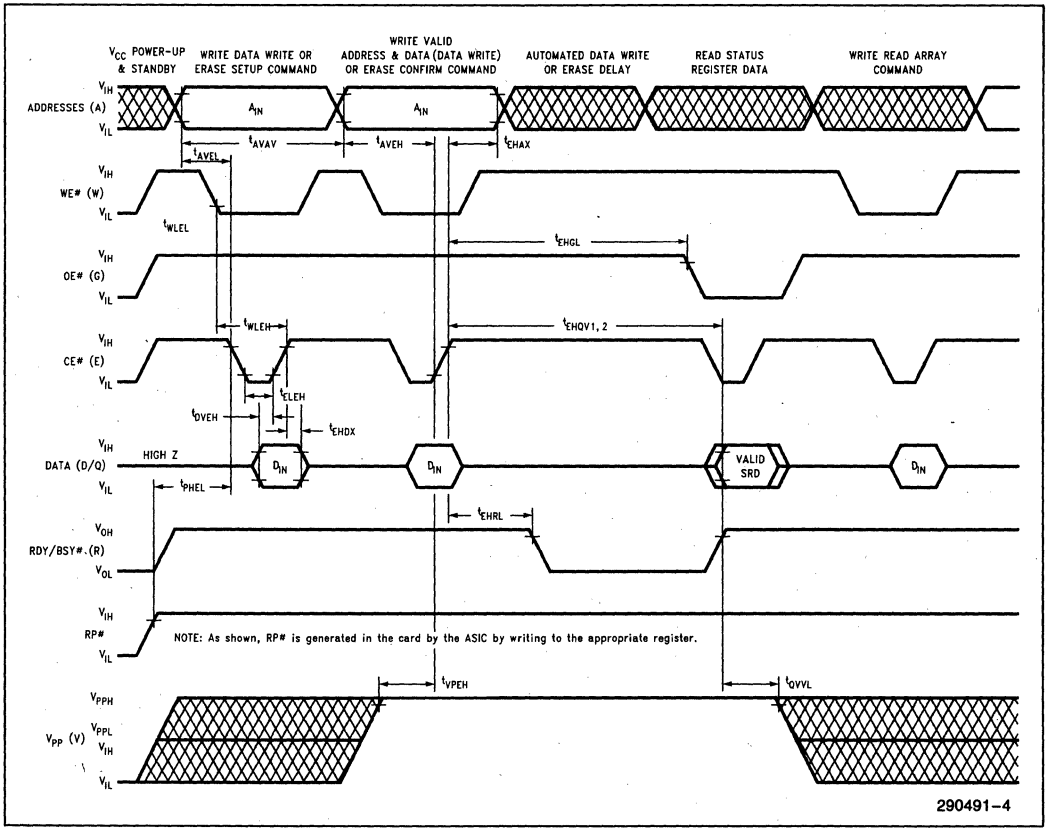


Figure 5. Alternate AC Waveform for Write Operations

10.4 Power-Up/Power-Down

Symbol PCMCIA	Parameter	Notes	Min	Max	
V_i (CE)	CE# Signal Level (0.0V < V_{CC} < 2.0V)	1	0	V_{iMAX}	V
	CE# Signal Level (2.0V < V_{CC} < V_{IH})	1	V_{CC} − 0.1	V_{iMAX}	V
	CE# Signal Level (V_{IH} < V_{CC})	1	V_{IH}	V_{iMAX}	V
t_{su} (V_{CC})	CE# Setup Time		20		ms
t_{su} (RESET)	CE# Setup Time		20		ms
t_{rec} (V_{CC})	CE# Recover Time		1.0		µs
t_{pr}	V_{CC} Rising Time	2	0.1	300	ms
t_{pf}	V_{CC} Falling Time	2	3.0	300	ms
t_w (RESET)	RESET Width		10		µs
t_h (Hi-Z Reset)	RESET Width		1		ms
t_s (Hi-Z Reset)	RESET Width		0		ms

NOTES:
1. V_{iMAX} means Absolute Maximum Voltage for input in the period of 0.0V < V_{CC} < 2.0V, V_i (CE#) is only 0.00V ~ V_{iMAX}.
2. The t_{pr} and t_{pf} are defined as "linear waveforms" in the period of 10% to 90%, or vice-versa. Even if the waveform is not a "linear waveform," its rising and falling time must meet this specification.

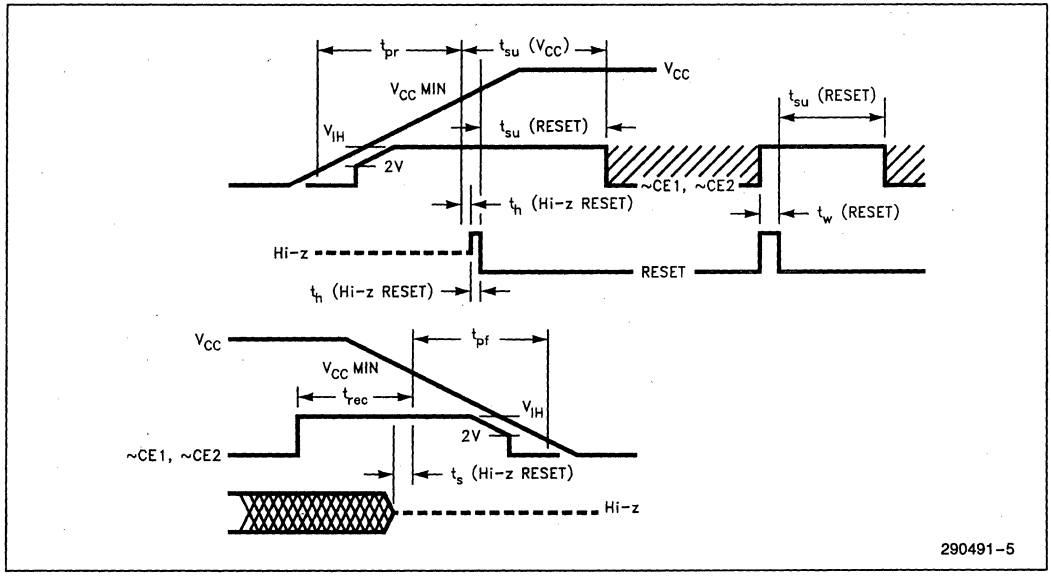


Figure 6. Power-Up/Down Timing for Systems Supporting RESET

10.5 Series 2+ Card Information Structure Tuple Changes

Address	Value	Description
06H	22H	150 ns
	32H	250 ns
14H	32H	250 ns
150H	12H	4 MB - 150 ns
	22H	8 MB - 150 ns
	42H	20 MB - 200 ns

11.0 CAPACITANCE

T_A = 25°C, f = 1.0 MHz

Symbol	Pins	Typ	Max	Unit
C_{IN}	A_0	15	30	pF
C_{IN}	Address/Control	10	20	pF
C_{IN}	V_{CC}, V_{PP}	2	2	μF
C_{OUT}	Output	10	20	pF

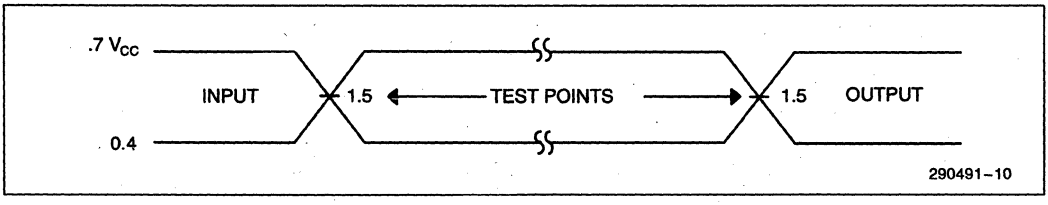


Figure 7. Transient Input/Output Reference Waveform (V_{CC} = 5.0V) for Standard Test Configuration

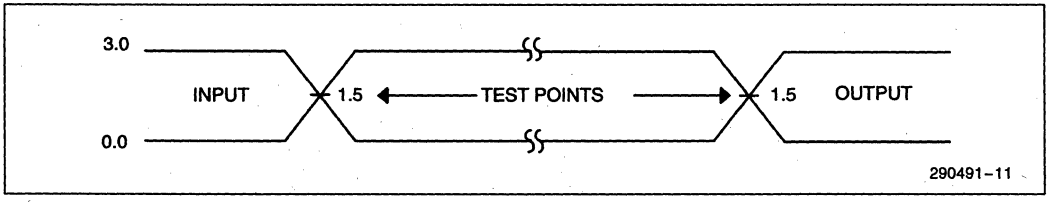


Figure 8. Transient Input/Output Reference Waveform (V_{CC} = 3.3V) for Standard Test Configuration

12.0 ERASE AND DATA WRITE PERFORMANCE[1,3]

V_{CC} = 3.3V ± 0.3V, T_A = 0°C to +70°C

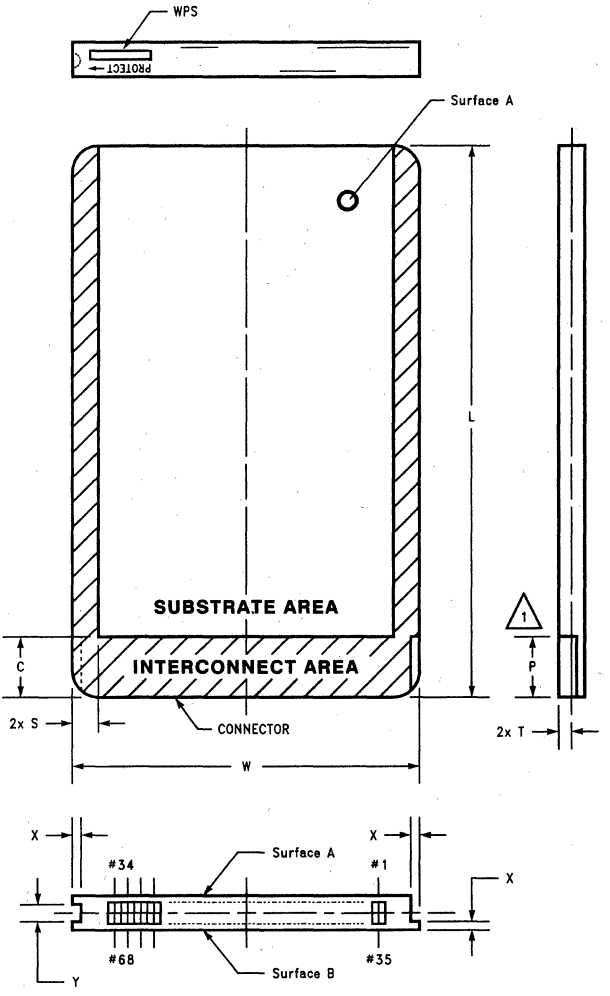
Symbol	Parameter	Notes	Min	Typ[1]	Max	Units	Test Conditions
	Page Buffer Word Write Time	2		2.2		µs	
t_{WHRH1}	Word/Byte Write Time	2		9 µs	3 ms		
t_{WHRH2}	Block Write Time	2		0.6	2.1	sec	Byte Write Mode
	Block Erase Time	2		0.8	10	sec	
	Full Chip Erase Time	2		51.2		sec	

V_{CC} = 5.0V ± 0.5V, T_A = 0°C to +70°C

Symbol	Parameter	Notes	Min	Typ[1]	Max	Units	Test Conditions
	Page Buffer Word Write Time	2		2.1		µs	
t_{WHRH1}	Word/Byte Write Time	2,4		6 µs	3 ms		
t_{WHRH2}	Block Write Time	2		0.4	2.1	sec	Byte Write Mode
	Block Erase Time	2		0.6	10	sec	
	Full Chip Erase Time	2		38.4		sec	

NOTES:
1. 25°C, and normal voltages.
2. Excludes system-level overhead.
3. These performance numbers are valid for all speed versions.
4. To maximize system performance, the RDY/BSY# signal should be polled instead of using the maximum Word/Byte write time as a delay timer.
The maximum Word/Byte write time is the absolute maximum time it takes the write algorithm to complete. The majority of the bits program in the typical value specified above.

13.0 PACKAGING



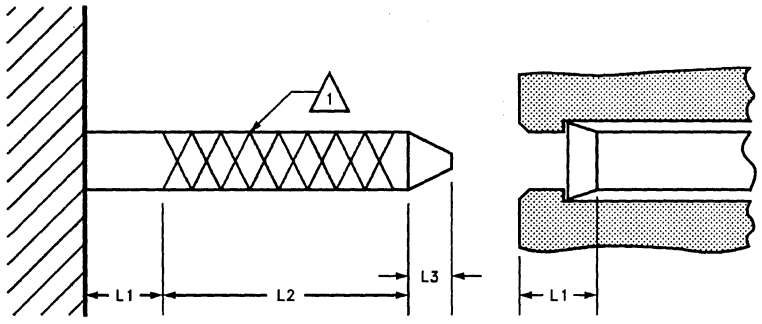
C MIN	L ± 0.008	P MIN ⚠1	S MIN	T ⚠2	W ± 0.004	X ± 0.002	Y ± 0.002
0.294 (10.0)	3.370 (85.60)	0.394 (10.0)	0.118 (3.0)	0.065 (1.65)	2.126 (54.0)	0.039 (1.00)	0.063 (1.60)

⚠1 POLARIZATION KEY LENGTH.

⚠2 INTERCONNECT AREA TOLERANCE = ±0.002
SUBSTRATE AREA TOLERANCE = ±0.004

3 MILLIMETERS ARE IN PARENTHESIS ().

290491-6



PIN SOCKET CONTACT

1. PIN/SOCKET CONTACT AREA
2. MILLIMETERS ARE IN PARENTHESIS ()

290491-7

6

14.0 ADDITIONAL INFORMATION

14.1 Ordering Information

iMC020FLSP,SBXXXXX

WHERE:

i	= INTEL
MC	= MEMORY CARD
020	= DENSITY IN MEGABYTES (004,020 AVAILABLE)
FL	= FLASH TECHNOLOGY
S	= BLOCKED ARCHITECTURE
P	= PERFORMANCE
SBXXXXX	= CUSTOMER IDENTIFIER

14.2 References

Order Number	Document
290434	Series 2+ Flash Memory Card Datasheet
297373	"Series 2+ Flash Memory Card User's Manual"
290489	"28F016SA 16-Mbit (1 Mb x 16, 2 Mb x 8) FlashFile™ Memory" Datasheet
290429	"28F08SA 8-Mbit (1 Mb x 8) FlashFile™ Memory" Datasheet
292092	AP-357, "Power Supply Solutions for Flash Memory"
292097	AP-362, "Implementing Mobile PC Designs Using High Density FlashFile™ Components"
292126	AP-377, "The 28F016SA Software Drivers"
292127	AP-378, "Enhanced Features of the 28F016SA"
292158	AP-606, "Interchangablility of Series 1, Series 2, and Series 2+ Flash Memory Cards"
294015	ER-31, "The 28F016SA FlashFile™ Memory"
294016	ER-33, "ETOX™ IV Flash Memory Technology: Insight to Intel's Fourth Generation Process Innovation"

14.3 Revision History

Number	Description
001	Original Version
002	Page Buffer Write to Flash Command Code Correction (from 08H to 0CH). Series 2+ Tuples and AC Characteristics Tables Include Support for 150 ns Access.
003	TTL DC Characteristics Tables Added. General DC Characteristics Table Changed to Reflect TTL Levels.
004	AC Characteristics Condensed to Include V_{PP} Pump Information.
005	Series 2+ 8-Meg and 40-Meg Cards Added to Datasheet. Table of Contents Numbering System Added. Component Management Register Tables Reformatted. I_{CCR} Values Increased for CMOS and TTL. 3.3V Timings Updated to 250 ns.
006	Changed I_{BB} S and SL max values

SERIES 2 FLASH MEMORY CARDS
iMC002FLSA, iMC004FLSA, iMC010FLSA, iMC020FLSA

Extended Temperature Specifications Included

- **2, 4, 10 and 20 Megabyte Capacities**
- **PCMCIA 2.1/JEIDA 4.1 68-Pin Standard**
  - **— Hardwired Card Information Structure**
  - **— Byte- or Word-Wide Selectable**
- **Component Management Registers for Card Status/Control and Flexible System Interface**
- **Automatic Erase/Write**
  - **— Monitored with Ready/Busy Output**
- **Card Power-Down Modes**
  - **— Deep-Sleep for Low Power Applications**
- **Mechanical Write Protect Switch**
- **Solid-State Reliability**
- **Intel FlashFile™ Architecture**
- **High-Performance Read Access**
  - **— 200 ns Maximum**
- **High-Performance Random Writes**
  - **— 10 μs Typical Word Write**
- **Erase Suspend to Read Command**
  - **— Keeps Erase as Background Task**
- **Nonvolatility (Zero Retention Power)**
  - **— No Batteries Required for Back-up**
- **ETOX™ III 0.8μ Flash Memory Technology**
  - **— 5V Read, 12V Erase/Write**
  - **— High-Volume Manufacturing Experience**
- **Extended Temperature Version**
  - **— −40°C to +85°C**

Intel's Series 2 Flash Memory Card facilitates high-performance disk emulation in mobile PCs and dedicated equipment. Manufactured with Intel's ETOX™ III 0.8μ, FlashFile Memory devices, the Series 2 Card allows code and data retention while erasing and/or writing other blocks. Additionally, the Series 2 Flash Memory Card features low power modes, flexible system interfacing and a 200 ns read access time. When coupled with Intel's low-power microprocessors, these cards enable high-performance implementations of mobile computers and systems.

Series 2 Cards conform to the Personal Computer Memory Card International Association (PCMCIA 2.1)/Japanese Electronics Industry Development Association (JEIDA 4.1) 68-pin standard, providing electrical and physical compatibility.

Data file management software, such as Microsoft's* Flash File System (FFS), provide data file storage and memory management, much like a disk operating system. Intel's Series 2 Flash Memory Cards, coupled with flash file management software, effectively provide a removable, all-silicon mass storage solution with higher performance and reliability than disk-based memory architectures.

Designing with Intel's FlashFile Architecture enables OEM system manufacturers to design and manufacture a new generation of mobile PCs and dedicated equipment where high performance, ruggedness, long battery life and lighter weight are a requirement. For large user groups in workstation environments, the Series 2 Cards provide a means to securely store user data and backup system configuration/status information.

ETOX, FlashFile, and i386SL are trademarks of Intel Corporation.
Microsoft is a trademark of Microsoft Corporation.

December 1994
Order Number: 290434-004

Table 1. Series 2 Flash Memory Card Pinout

Pin	Signal	I/O	Function	Active
1	GND		Ground	
2	DQ_3	I/O	Data Bit 3	
3	DQ_4	I/O	Data Bit 4	
4	DQ_5	I/O	Data Bit 5	
5	DQ_6	I/O	Data Bit 6	
6	DQ_7	I/O	Data Bit 7	
7	CE_1#	I	Card Enable 1	LO
8	A_{10}	I	Address Bit 10	
9	OE#	I	Output Enable	LO
10	A_{11}	I	Address Bit 11	
11	A_9	I	Address Bit 9	
12	A_8	I	Address Bit 8	
13	A_{13}	I	Address Bit 13	
14	A_{14}	I	Address Bit 14	
15	WE#	I	Write Enable	LO
16	RDY/BSY#		Ready-Busy	HI/LO
17	V_{CC}		Supply Voltage	
18	V_{PP1}		Supply Voltage	
19	A_{16}	I	Address Bit 16	
20	A_{15}	I	Address Bit 15	
21	A_{12}	I	Address Bit 12	
22	A_7	I	Address Bit 7	
23	A_6	I	Address Bit 6	
24	A_5	I	Address Bit 5	
25	A_4	I	Address Bit 4	
26	A_3	I	Address Bit 3	
27	A_2	I	Address Bit 2	
28	A_1	I	Address Bit 1	
29	A_0	I	Address Bit 0	
30	DQ_0	I/O	Data Bit 0	
31	DQ_1	I/O	Data Bit 1	
32	DQ_2	I/O	Data Bit 2	
33	WP	O	Write Protect	HI
34	GND		Ground	

Pin	Signal	I/O	Function	Active
35	GND		Ground	
36	CD_1#	O	Card Detect 1	LO
37	DQ_{11}	I/O	Data Bit 11	
38	DQ_{12}	I/O	Data Bit 12	
39	DQ_{13}	I/O	Data Bit 13	
40	DQ_{14}	I/O	Data Bit 14	
41	DQ_{15}	I/O	Data Bit 15	
42	CE_2#	I	Card Enable 2	LO
43	VS_1	O	Voltage Sense 1	N.C.
44	RFU		Reserved	
45	RFU		Reserved	
46	A_{17}	I	Address Bit 17	
47	A_{18}	I	Address Bit 18	
48	A_{19}	I	Address Bit 19	
49	A_{20}	I	Address Bit 20	
50	A_{21}	I	Address Bit 21	
51	V_{CC}		Supply Voltage	
52	V_{PP2}		Supply Voltage	
53	A_{22}	I	Address Bit 22	
54	A_{23}	I	Address Bit 23	
55	A_{24}	I	Address Bit 24	
56	A_{25}		No Connect	
57	VS_2	O	Voltage Sense 2	N.C.
58	RST	I	Reset	HI
59	WAIT#	O	Extend Bus Cycle	LO
60	RFU		Reserved	
61	REG#	I	Register Select	LO
62	BVD_2	O	Batt. Volt Det 2	
63	BVD_1	O	Batt. Volt Det 1	
64	DQ_8	I/O	Data Bit 8	
65	DQ_9	I/O	Data Bit 9	
66	DQ_{10}	I/O	Data Bit 10	
67	CD_2#	O	Card Detect 2	LO
68	GND		Ground	

Table 2. Series 2 Flash Memory Card Pin Descriptions

Symbol	Type	Name and Function
A_0–A_{25}	I	**ADDRESS INPUTS:** A_0 through A_{25} are address bus lines which enable direct addressing of 64 megabytes of memory on a card. A_0 is not used in word access mode. A_{24} is the most significant address bit. Note: A_{25} is a no-connect but should be provided on host side.
DQ_0–DQ_{15}	I/O	**DATA INPUT/OUTPUT:** DQ_0 through DQ_{15} constitute the bidirectional data bus. DQ_{15} is the most significant bit.
CE_1#, CE_2#	I	**CARD ENABLE 1, 2:** CE_1# enables even bytes, CE_2# enables odd bytes. Multiplexing A_0, CE_1# and CE_2# allows 8-bit hosts to access all data on DQ_0 through DQ7. (See Table 3 for a more detailed description.)
OE#	I	**OUTPUT ENABLE:** Active low signal gating read data from the memory card.
WE#	I	**WRITE ENABLE:** Active low signal gating write data to the memory card.
RDY/BSY#	O	**READY/BUSY OUTPUT:** Indicates status of internally timed erase or write activities. A high output indicates the memory card is ready to accept accesses. A low output indicates that a device(s) in the memory card is(are) busy with internally timed activities. See text for an alternate function (READY-BUSY MODE REGISTER).
CD_1# & CD_2#	O	**CARD DETECT 1, 2:** These signals provide for correct card insertion detection. They are positioned at opposite ends of the card to detect proper alignment. The signals are connected to ground internally on the memory card and will be forced low whenever a card is placed in the socket. The host socket interface circuitry shall supply 10K or larger pull-up resistors on these signal pins.
WP	O	**WRITE PROTECT:** Write Protect reflects the status of the Write-Protect switch on the memory card. WP set high = write protected, providing internal hardware write lockout to the flash array.
V_{PP1}, V_{PP2}		**WRITE/ERASE POWER SUPPLY:** (12V nominal) for erasing memory array blocks or writing data in the array. They must be 12V to perform an erase/write operation. V_{PP1} supplies even byte Erase/Write voltage and V_{PP2} supplies the odd byte Erase/Write voltage.
V_{CC}		**CARD POWER SUPPLY** (5V nominal) for all internal circuitry.
GND	I	**GROUND** for all internal circuitry.
REG#	I	**REGISTER SELECT** provides access to Series 2 Flash Memory Card registers and Card Information Structure in the Attribute Memory Plane.
RST	I	**RESET** from system, active high. Places card in Power-On Default State. RESET pulse width must be ≥ 200 ns.
WAIT#	O	**WAIT (Extend Bus Cycle)** is used by Intel's I/O cards and is driven high.
BVD_1, BVD_2	O	**BATTERY VOLTAGE DETECT:** Upon completion of the power on reset cycle, these signals are driven high to maintain SRAM-card compatibility.
RFU		**RESERVED FOR FUTURE USE**
NC		**NO INTERNAL CONNECTION**. Pin may be driven or left floating.
VS_1, VS_2		**VOLTAGE SENSE:** Notifies the host Socket of the card's V_{CC} requirements. VS_1 and VS_2 are both open, indicating a 5V V_{CC} card.

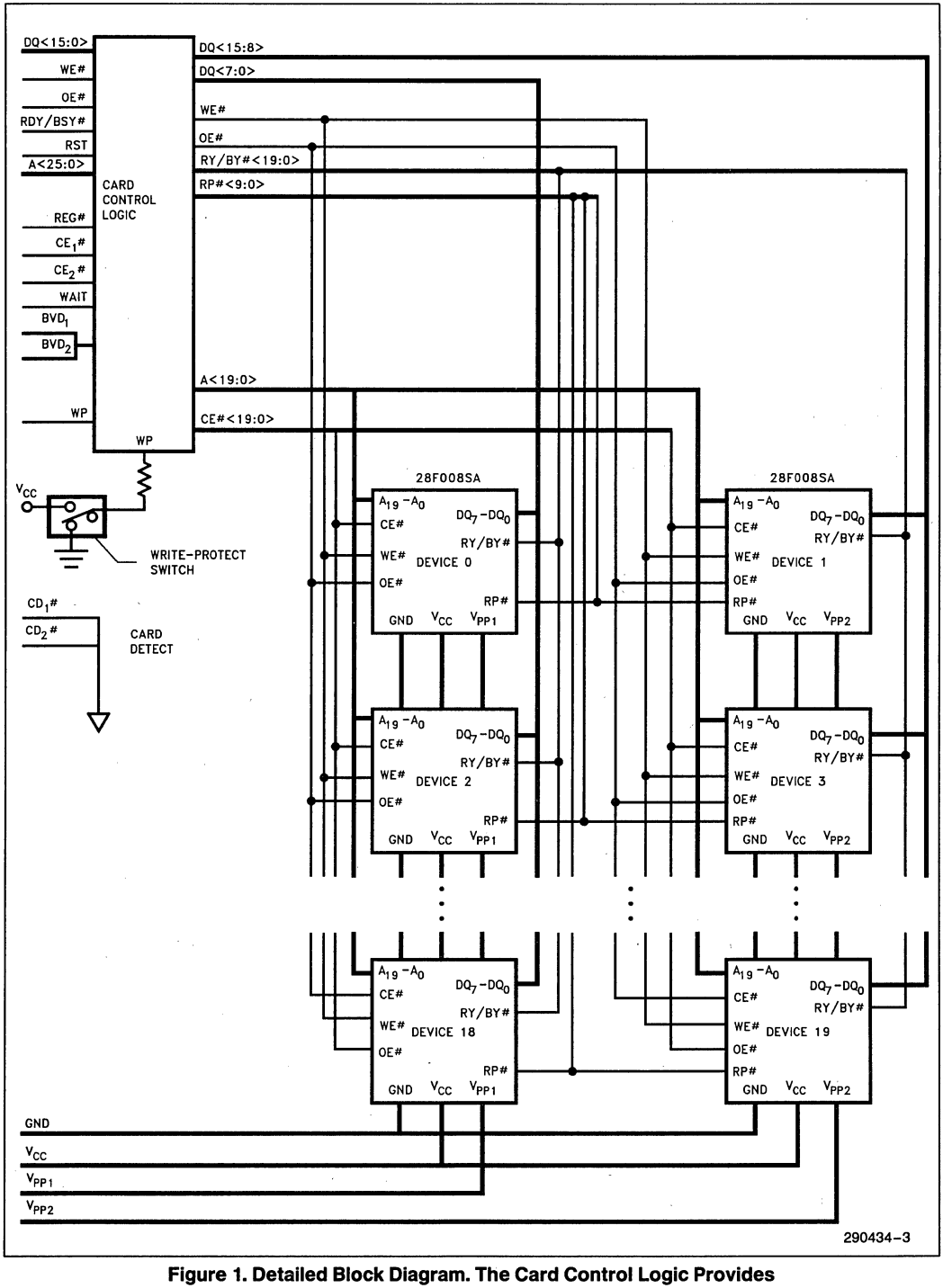


Figure 1. Detailed Block Diagram. The Card Control Logic Provides Decoding Buffering and Control Signals.

APPLICATIONS

Intel's second generation Series 2 Flash Memory Cards facilitate high performance disk emulation for the storage of data files and application programs on a purely solid-state removable medium. File management software, such as Microsoft's Flash File System, in conjunction with the Series 2 Flash Memory Cards enable the design of high-performance light-weight notebook, palmtop, and pen-based PCs that have the processing power of today's desktop computers.

Application software stored on the flash memory card substantially reduces the slow disk-to-DRAM download process. Replacing the mechanical disk results in a dramatic enhancement of read performance and substantial reduction of power consumption, size and weight—considerations particularly important in portable PCs and equipment. The Series 2 Card's high performance read access time allows the use of Series 2 Cards in an "execute-in-place" (XIP) architecture. XIP eliminates redundancy associated with DRAM/Disk memory system architectures. Operating systems stored in Flash Memory decreases system boot or program load times, enabling the design of PCs that boot, operate, store data files and execute application programs from/to nonvolatile memory without losing the ability to perform an update.

File management systems modify and store data files by allocating flash memory space intelligently. Wear leveling algorithms, employed to equally distribute the number of rewrite cycles, ensure that no particular block is cycled excessively relative to other blocks. This provides hundreds of thousands of hours of power on usage.

This file management software enables the user to interact with the flash memory card in precisely the same way as a magnetic disk.

For example, the Microsoft Flash File System enables the storage and modification of data files by utilizing a linked-list directory structure that is evenly distributed along with the data throughout the memory array. The linked-list approach minimizes file fragmentation losses by using variable-sized data structures rather than the standard sector/cluster method of disk-based systems.

Series 2 Flash Memory Cards provide durable nonvolatile memory storage for mobile PCs on the road, facilitating simple transfer back into the desktop environment.

For systems currently using a static RAM/battery configuration for data acquisition, the Series 2 Flash Memory Card's nonvolatility eliminates the need for battery backup. The concern for battery failure no longer exists, an important consideration for portable computers and medical instruments, both requiring continuous operation. Series 2 Cards consume no power when the system is off, and only 5 μA in Deep-Sleep mode (20 Megabyte card). Furthermore, Flash Memory Cards offer a considerable cost and density advantage over memory cards based on static RAM with battery backup.

Besides disk emulation, the Series 2 Card's electrical block-erasure, data writability, and inherent nonvolatility fit well with data accumulation and recording needs. Electrical block-erasure provides design flexibility to selectively rewrite blocks of data, while saving other blocks for infrequently updated parameters and lookup tables. For example, networks and systems that utilize large banks of battery-backed DRAM to store configuration and status benefit from the Series 2 Flash Card's nonvolatility and reliability.

SERIES 2 ARCHITECTURE OVERVIEW

The Series 2 Flash Memory Card contains a 2 to 20 Megabyte Flash Memory array consisting of 2 to 20 28F008SA FlashFile Memory devices. Each 28F008SA contains sixteen individually-erasable, 64 Kbyte blocks; therefore, the Flash Memory Card contains from 32 to 320 device blocks. It also contains two Card Control Logic devices that manage the external interface, address decoding, and component management logic. (Refer to Figure 1 for a block diagram.)

To support PCMCIA-compatible word-wide access, devices are paired so that each accessible memory block is 64 KWords (see Figure 2). Card logic allows the system to write or read one word at a time, or one byte at a time by referencing the high or low byte. Erasure can be performed on the entire block pair (high and low device blocks simultaneously), or on the high or low byte portion separately.

Also in accordance with PCMCIA specifications this product supports byte-wide operation, in which the flash array is divided into 128K x 8 bit device blocks. In this configuration, odd bytes are multiplexed onto the low byte data bus.

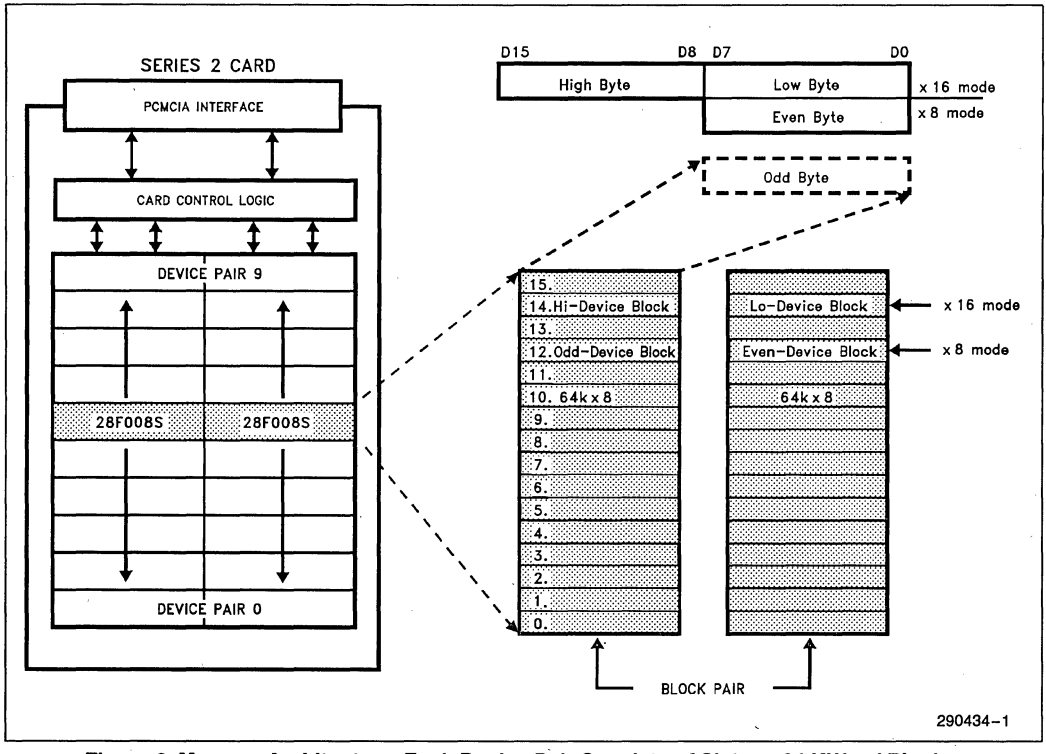


Figure 2. Memory Architecture. Each Device Pair Consists of Sixteen 64 KWord Blocks.

Series 2 Flash Memory Cards offer additional features over the Bulk Erase Flash Card product family (refer to iMC001FLKA, iMC002FLKA and iMC004FLKA data sheets). Some of the more notable enhancements include: high density capability, erase blocking, internal write/erase automation, erase suspension to read, Component Management Registers that provide software control of device-level functions and a deep-sleep mode.

Erase blocking facilitates solid-state storage applications by allowing selective memory reclamation. Multiple 64 Kbyte blocks may be simultaneously erased within the memory card as long as not more than one block per device is erasing. This shortens the total time required for erasure, but requires additional supply current. A block typically requires 1.6 seconds to erase. Each memory block can be erased and completely written 100,000 times.

Erase suspend allows the system to temporarily interrupt a block erase operation. This mode permits reads from alternate device blocks while that same device contains an erasing block. Upon completion of the read operation, erasure of the suspended block must be resumed.

Write/erase automation simplifies the system software interface to the card. A two-step command sequence initiates write or erase operations and provides additional data security. Internal device circuits automatically execute the algorithms and timings necessary for data-write or block-erase operations, including verifications for long-term data integrity. While performing either data-write or block-erase, the memory card interface reflects this by bringing its RDY/BSY# (Ready/Busy) pin low. This output goes high when the operation completes. This feature reduces CPU overhead and allows software polling or hardware interrupt mechanisms. Writing memory data is achieved in single byte or word increments, typically in 10 μs.

Read access time is 200 ns or less over the $-40°C$ to $+85°C$ temperature range.

The Reset-PowerDown mode reduces power consumption to less than 5 μA to help extend battery life of portable host systems. Activated through software control, this mode optionally affects the entire flash array (Global Reset-PowerDown Register) or specific device pairs (Sleep Control Register).

PCMCIA/JEIDA INTERFACE

The Series 2 Flash Memory Card interface supports the PCMCIA 2.1 and JEIDA 4.1 68-pin card format (see Tables 1 and 2). Detailed specifications are described in the PC Card Standard, Release 2.1, July, 1993, published by PCMCIA. The Series 2 Card conforms to the requirements of both Release 1 and Release 2 of the PC Card Standard.

Series 2 Card pin definitions are equivalent to the Bulk-Erase Flash Card except that certain No Connects are now used. A_{22} through A_{24}, RST (Reset), and RDY/BSY# (Ready/Busy) have pin assignments as set by the PCMCIA standard.

NOTE: The READY/BUSY signal is abbreviated as RDY/BSY# by PCMCIA (card level) and as RY/BY# by JEDEC (component level).

The outer shell of the Series 2 card meets all PCMCIA/JEIDA Type 1 mechanical specifications. See Figure 19 for mechanical dimensions.

WRITE PROTECT SWITCH

A mechanical write protect switch provides the card's memory array with internal write lockout. The Write-Protect (WP) output pin reflects the status of this mechanical switch. It outputs a high signal (V_{OH}) when writes are disabled. This switch does not lock out writes to the Component Management Registers.

BATTERY VOLTAGE DETECT

PCMCIA requires two signals, BVD_1 and BVD_2, be supplied at the interface to reflect card battery condition. Flash Memory Cards do not require batteries. When the power on reset cycle is complete, BVD_1 and BVD_2 are driven high to maintain compatibility.

CARD DETECT

Two signals, CD_1# and CD_2#, allow the host to determine proper socket seating. They reside at opposite ends of the connector and are tied to ground within the memory card.

DESIGN CONSIDERATIONS

The Series 2 Card consists of two separate memory planes: the Common Memory Plane (or Main Memory) and the Attribute Memory Plane. The Common Memory Plane resides in the banks of device pairs and represents the user-alterable memory space.

The Component Management Registers (CMR) and the hardwired Card Information Structure (CIS) reside in the Attribute Memory Plane within the Card Control Logic, as shown in Figure 3. The Card Control Logic interfaces the PCMCIA connector and the internal flash memory array and performs address decoding and data control.

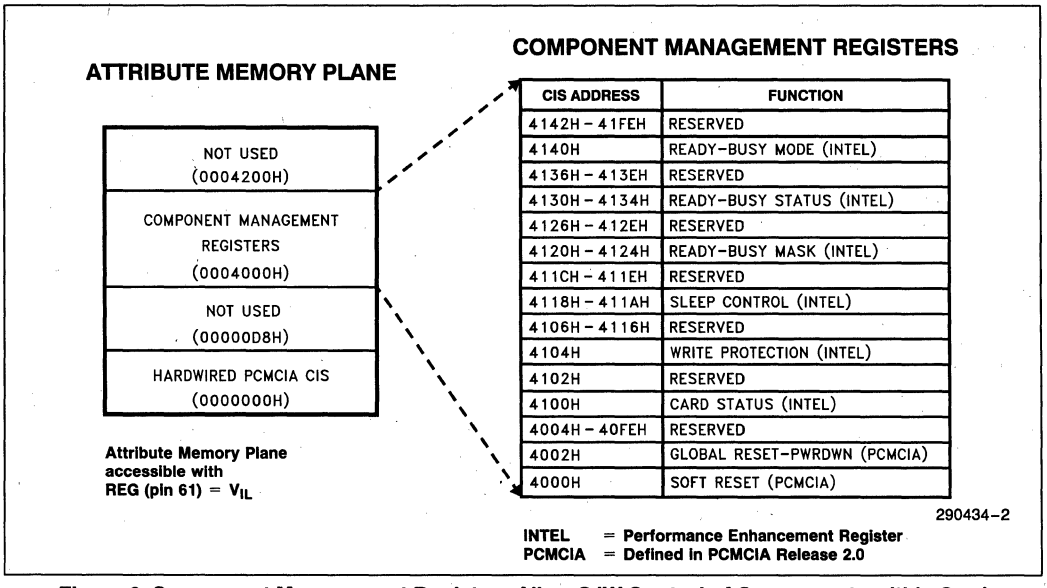


Figure 3. Component Management Registers Allow S/W Control of Components within Card

ADDRESS DECODE

Address decoding provides the decoding logic for the 2 to 20 Device Chip Enables and the elements of the Attribute Memory Plane. REG# selects between the Common Memory Plane (REG# = V_{IH}) and the Attribute Memory Plane (REG# = V_{IL}).

NOTE:
The Series 2 Card has *active* address inputs A_0 to A_{24} implying that reading and writing to addresses beyond 32 Megabytes causes wraparound. Furthermore, reads to illegal addresses (for example, between 20 and 32 Meg on a 20 Megabyte card) returns 0FFFFh data.

The 28F008SA devices, storing data, applications or firmware, form the Common Memory Plane accessed individually or as device pairs. Memory is linearly mapped in the Common Memory Plane. Three memory access modes are available when accessing the Common Memory Plane: Byte-Wide, Word Wide, and Odd-Byte modes.

Additional decoding selects the hardwired PCMCIA CIS and Component Management Registers mapped in the Attribute Memory Plane beginning at address 000000H.

The 512 memory-mapped even-byte CMRs are linearly mapped beginning at address 4000H in the Attribute Memory Plane.

DATA CONTROL

Data Control Logic selects the path and direction for accessing the Common or Attribute Memory Plane. It controls any of the PCMCIA-defined Word-Wide, Byte-Wide or Odd-Byte modes for either reads or writes to these areas. As shown in Table 3, input pins which determine these selections are REG#, A_0 through A_{24}, WE#, OE#, CE_1#, and CE_2#. PCMCIA specifications allow only even-byte access to the Attribute Memory Plane.

In Byte-Wide mode, bytes contiguous in software actually alternate between two device blocks of a device pair. Therefore, erasure of one device block erases every other contiguous byte. In accordance with the PCMCIA standard for memory configuration, the Series 2 Card does not support confining contiguous bytes within one flash device when in by-8 mode.

Table 3. Data Access Mode Truth Table

Function Mode	REG#	$CE\#_2$	$CE\#_1$	A_0	OE#	WE#	V_{PP2}	V_{PP1}	D_{15}–D_8	D_7–D_0
COMMON MEMORY PLANE										
STANDBY(1)	X	H	H	X	X	X	V_{PPL}(2)	V_{PPL}(2)	HIGH-Z	HIGH-Z
BYTE READ	H	H	L	L	L	H	V_{PPL}(2)	V_{PPL}(2)	HIGH-Z	EVEN-BYTE
	H	H	L	H	L	H	V_{PPL}(2)	V_{PPL}(2)	HIGH-Z	ODD-BYTE
WORD READ	H	L	L	X	L	H	V_{PPL}(2)	V_{PPL}(2)	ODD-BYTE	EVEN-BYTE
ODD-BYTE READ	H	L	H	X	L	H	V_{PPL}(2)	V_{PPL}(2)	ODD-BYTE	HIGH-Z
BYTE WRITE	H	H	L	L	H	L	V_{PPH}	V_{PPH}	X	EVEN-BYTE
	H	H	L	H	H	L	V_{PPH}	V_{PPH}	X	ODD-BYTE
WORD WRITE	H	L	L	X	H	L	V_{PPH}	V_{PPH}	ODD-BYTE	EVEN-BYTE
ODD-BYTE WRITE	H	L	H	X	H	L	V_{PPH}	V_{PPL}(2)	ODD-BYTE	X
ATTRIBUTE MEMORY PLANE										
BYTE READ	L	H	L	L	L	H	X(2)	X(2)	HIGH-Z	EVEN-BYTE
	L	H	L	H	L	H	X(2)	X(2)	HIGH-Z	INVALID
WORD READ	L	L	L	X	L	H	X(2)	X(2)	INVALID DATA(3)	EVEN-BYTE
ODD-BYTE READ	L	L	H	X	L	H	X(2)	X(2)	INVALID DATA(3)	HIGH-Z
BYTE WRITE	L	H	L	L	H	L	X(2)	X(2)	X	EVEN-BYTE
	L	H	L	H	H	L	X(2)	X(2)	X	INVALID OPERATION(3)
WORD WRITE	L	L	L	X	H	L	X(2)	X(2)	INVALID OPERATION(3)	EVEN-BYTE
ODD-BYTE WRITE	L	L	H	X	H	L	X(2)	X(2)	INVALID OPERATION(3)	X

NOTES:
1. Standby mode is valid in Common Memory or Attribute Memory access.
2. To meet the low power specifications, $V_{PP} = V_{PPL}$; however V_{PPH} presents no reliability problems.
3. Odd-Byte data are not valid during access to the Attribute Memory Plane.
4. H = V_{IH}, L = V_{IL}, X = Don't Care.

PRINCIPLES OF OPERATION

Intel's Series 2 Flash Memory Card provides electrically-alterable, non-volatile, random-access storage. Individual 28F008SA devices utilize a Command User Interface (CUI) and Write State Machine (WSM) to simplify block-erasure and data write operations.

COMMON MEMORY ARRAY

Figure 4 shows the Common Memory Plane's organization. The first block pair (64 KWords) of Common Memory, referred to as the Common Memory Card Information Structure Block, *optionally* extends the hardwired CIS in the Attribute Memory Plane for additional card information. This may be written during initial card formatting for OEM customization. Since this CIS Block is part of Common Memory, its data can be altered. Write access to the Common Memory CIS Block is controlled by the Write Protect Control Register which may be activated by system software after power-up. Additionally, the entire Common Memory plane (minus the Common Memory CIS Block) may be software write protected. *Note that the Common Memory CIS Block is not part of the Attribute Memory Plane. Do not assert REG# to access the Common Memory CIS Block.*

13FFFFFH 1200000H	Device Pair 9
1000000H	Device Pair 8
0E00000H	Device Pair 7
0C00000H	Device Pair 6
0A00000H	Device Pair 5
0800000H	Device Pair 4
0600000H	Device Pair 3
0400000H	Device Pair 2
0200000H	Device Pair 1
0020000H	Device Pair 0
0000000H	Optional CIS

Figure 4. Common Memory Plane. Use the Optional Common Memory Plane CIS for Custom Card Format Information.

HARDWIRED CIS

The card's structure description resides in the even-byte locations starting at 0000H and going to the CIS ending tuple (FNULL) within the Attribute Memory Plane. Data included in the hardwired CIS consists of tuples. Tuples are a variable-length list of data blocks describing details such as manufacturer's name, the size of each memory device and the number of flash devices within the card.

COMPONENT MANAGEMENT REGISTERS (CMRs)

The CMRs in the Attribute Memory Plane provide special, software-controlled functionality. Card Control Logic includes circuitry to access the CMRs. REG (PCMCIA, pin 61) selects the Attribute Memory Plane (and therefore the CMRs) when equal to V_{IL}.

CMRs are classified into two categories: those defined by PCMCIA R2.1 and those included by Intel (referred to as Performance Enhancement Registers) to enhance the interface between the host system and the card's flash memory array. CMRs (See Figure 3) provide seven control functions—Ready-Busy Interrupt Mode, Device Ready-Busy Status, Device Ready-Busy Mask, Reset-PowerDown Control, Software-controlled Write Protection, Card Status and Soft Reset.

SOFT RESET REGISTER (PCMCIA)

(CONFIGURATION OPTION)

The SOFT RESET REGISTER (Attribute Memory Plane Address 4000H, Figure 5) is defined in the PCMCIA Release 2.0 specification as the Configuration Option Register.

Bit 7 is the soft reset bit (SRESET). Writing a 1 to this bit initiates card reset to the power-on default state (see Side Bar page 11). This bit must be cleared to use the CMRs or to access the devices.

SRESET implements in software what the reset pin implements in hardware. On power-up, the card automatically assumes default conditions. Similar to the reset pin (pin 58), this bit clears at the end of a power-on reset cycle or a system reset cycle.

Bits 0 through 6 are not used by this memory card, but power up as zeroes for PCMCIA compatibility.

SOFT RESET REGISTER

(CONFIGURATION OPTION REGISTER)

(Read/Write Register)

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 4	BIT 2	BIT 1	BIT 0
4000H	SRESET	PCMCIA CONFIGURATION INDEX RESETS TO ZERO ON POWER UP						

1 = RESET, CLEAR TO ACCESS CARD

Figure 5. SOFT RESET REGISTER (PCMCIA). Sets the Memory Card in the Power-On Default State.

POWER-ON DEFAULT CONDITIONS

- All Devices Powered Up In Standby Mode
- Common Memory Available For Writes
- All Device Ready/Busy Outputs Unmasked
- PCMCIA Ready/Busy Mode Enabled
- Ready/Busy Output Goes To Ready

Global PowerDown Register (PCMCIA)

(Configuration and Status)

The Global Reset-PowerDown Register (Attribute Memory Plane Address 4002H, Figure 6) is referred to as the Configuration and Status Register in the PCMCIA Release 2.0 specification.

Bit 2 (RP) controls global card power-down. Writing a 1 to this bit places each device within the card into "Deep-Sleep" mode. *Devices in Deep-Sleep are not accessible.* Recovery from power-down requires 500 ns for reads and 1 μs for writes.

The RP bit defaults to 0 on card power-up or reset. Setting or clearing this bit has no affect on the bit settings of the Sleep Control Register.

The remaining Global Reset-PwrDwn Register bits are defined for Intel's family of I/O cards and are driven low for compatibility.

GLOBAL RESET-POWER-DOWN REGISTER

(CONFIGURATION AND STATUS REGISTER)

(Read/Write Register)

1 = POWER DOWN

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4002H	ZEROES					RP	ZEROES	

Figure 6. GLOBAL RESET-PWRDWN REGISTER (PCMCIA). The RP Bit Enables Reset PowerDown of All Flash Memory Devices.

CARD STATUS REGISTER

(Read Only Register)

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4100H	ADM	ADS	SRESET	CMWP	RP	CISWP	WP	RDY/BSY#

Figure 7. CARD STATUS REGISTER (Intel) Provides a Quick Review of the Card's Status

CARD STATUS REGISTER (INTEL)

The Read-Only, CARD STATUS REGISTER (Attribute Memory Plane Address 4100H, Figure 7) returns generalized status of the Series 2 Card and its CMRs.

Bit 0 (RDY/BSY#) reflects the card's RDY/BSY# (Ready-Busy) output. Software polling of this bit provides data-write or block-erase operation status. A zero indicates a busy device(s) in the card.

Bit 1 (WP) reports the position of the card's Write Protection switch with 1 indicating write protected. It reports the status of the WP pin.

Bit 2 (CISWP) reflects whether the Common Memory CIS is write protected using the WRITE PROTECT REGISTER, with 1 indicating write protected.

Bit 3 (RP) reports whether the entire flash memory array is in "Deep-Sleep" (Reset-PwrDwn) mode, with 1 indicating "Deep-Sleep". This bit reflects the RP bit of the GLOBAL RESET-POWERDOWN REGISTER. Powering down *all* device pairs individually (using the Sleep Control Register), also sets this bit.

Bit 4 (CMWP) reports whether the Common Memory Plane (minus Common Memory CIS) is write protected via the WRITE PROTECT REGISTER with 1 indicating write protected.

Bit 5 (SRESET) reflects the SRESET bit of the SOFT RESET REGISTER. It reports that the card is in Soft Reset with 1 indicating reset. When this bit is zero, the flash memory array and CMRs may be accessed, otherwise clear it via the SRESET REGISTER.

Bit 6 (ADS, ANY DEVICE SLEEP) is the "ORed" value of the SLEEP CONTROL REGISTER. Powering down any device pair sets this bit.

Bit 7 (ADM, ANY DEVICE MASKED) is the "ORed" value of the READY/BUSY MASK REGISTER. Masking any device sets this bit.

WRITE PROTECTION REGISTER (INTEL)

The WRITE PROTECTION REGISTER (Attribute Memory Plane Address 4104H, Figure 8) selects whether the optional Common Memory CIS and the remaining Common Memory blocks are write protected (see Figure 4).

Enable Common Memory CIS write protection by writing a 1 to the CISWP Bit (bit 0).

Enable write protection of the remaining Common Memory blocks by writing a 1 to the CMWP Bit (bit 1).

In the power-on default state, both bits are 0, and therefore not write protected.

Reserved bits (2–7) have undefined values and should be written as zeroes for future compatibility.

WRITE PROTECTION REGISTER

(Read/Write Register)

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4104H	RESERVED FOR FUTURE USE						CMWP	CISWP

1 = WRITE PROTECT

Figure 8. WRITE PROTECTION REGISTER (Intel) Eliminates Accidental Data Corruption

SLEEP CONTROL REGISTER (INTEL)

Unlike the GLOBAL RESET-POWERDOWN REGISTER, which simultaneously resets and places all flash memory devices into a Deep-Sleep mode, the SLEEP CONTROL REGISTER (Attribute Memory Plane Address 4118H–411AH, Figure 9) allows selective power-down control of individual device pairs.

Writing a 1 to a specific bit of the SLEEP CONTROL REGISTER places the corresponding device pair into the "Deep-Sleep" mode. *Devices in Deep-Sleep are not accessible.* On cards with fewer than 20 Megabytes (10 device pairs), writing a one to an absent device pair has no affect and reads back as zero.

This register contains all zeroes (i.e., not in Deep-Sleep mode) when the card powers up or after a hard or soft reset. Furthermore, the Global Reset-PowerDown Register has no affect on the contents of this register. Therefore, any bit settings of the Sleep Control Register will remain unchanged after returning from a global reset and power down (writing a zero to the RP bit of the Global Reset-Power-Down Register).

READY-BUSY STATUS REGISTER (INTEL)

The bits in the Read-only, READY-BUSY Status Register (Attribute Memory Plane Address 4130H-4134H, Figure 10) reflect the status (READY = 1, BUSY = 0) of each device's RY/BY# output. A busy condition indicates that a device is currently processing a data-write or block-erase operation.

These bits are logically "AND-ed" to form the Ready/Busy output (RDY/BSY#, pin 16) of the PCMCIA interface. On memory cards with fewer than 20 devices, unused Device RY/BY# Status Register bits appear as ready.

SLEEP CONTROL REGISTER

(Read/Write Register)

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
411AH	RESERVED						DEVICES 18/19	DEVICES 16/17
4118H	DEVICES 14/15	DEVICES 12/13	DEVICES 10/11	DEVICES 8/9	DEVICES 6/7	DEVICES 4/5	DEVICES 2/3	DEVICES 0/1

1 = SELECTED DEVICE PAIR IN POWER-DOWN MODE AND RESET

Figure 9. SLEEP CONTROL REGISTER (Intel) Allows Specific Devices to be Reset and Put into Power-Down Mode

READY-BUSY STATUS REGISTER

(Read/Write Register)

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4134H	RESERVED				DEVICE 19	DEVICE 18	DEVICE 17	DEVICE 16
4132H	DEVICE 15	DEVICE 14	DEVICE 13	DEVICE 12	DEVICE 11	DEVICE 10	DEVICE 9	DEVICE 8
4130H	DEVICE 7	DEVICE 6	DEVICE 5	DEVICE 4	DEVICE 3	DEVICE 2	DEVICE 1	DEVICE 0

1 = DEVICE READY, 0 = DEVICE BUSY

Figure 10. READY-BUSY STATUS REGISTER (Intel) Provides Operation Status of All Flash Memory Devices

READY-BUSY MASK REGISTER (INTEL)

The bits of the Read/Write READY-BUSY MASK REGISTER (Attribute Memory Plane Address 4120H–4124H, Figure 11) mask out the corresponding "AND-ed" READY-BUSY STATUS REGISTER bits from the PCMCIA data bus (RDY/BSY#, pin 16) and the CARD STATUS REGISTER RDY/BSY# Bit (bit 0).

In an unmasked condition (MASK REGISTER bits = 0), any device RY/BY# output going low pulls the card's RDY/BSY# output to V_{IL} (BUSY). In this case, all devices must be READY to allow the card's RDY/BSY# output to be ready (V_{IH}). *This is referred to as the PCMCIA READY-BUSY MODE. An alternate type of READY-BUSY function is described in the next section, READY-BUSY MODE REGISTER.*

READY-BUSY MASK

(Read/Write Register)

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4124H	RESERVED				DEVICE 19	DEVICE 18	DEVICE 17	DEVICE 16
4122H	DEVICE 15	DEVICE 14	DEVICE 13	DEVICE 12	DEVICE 11	DEVICE 10	DEVICE 9	DEVICE 8
4120H	DEVICE 7	DEVICE 6	DEVICE 5	DEVICE 4	DEVICE 3	DEVICE 2	DEVICE 1	DEVICE 0

1 = MASK ENABLED

Figure 11. READY-BUSY MASK REGISTER (Intel) Essential for Write Optimization

If the READY-BUSY MASK REGISTER bits are set to ones (masked condition), the RDY/BSY# output and the CARD STATUS REGISTER RDY/BSY# bit will reflect a READY condition regardless of the state of the corresponding devices. The READY-BUSY MASK REGISTER does not affect the READY-BUSY STATUS REGISTER allowing software polling to determine operation status.

Unmasked is the default condition for the bits in this register. On memory cards with fewer than 20 devices, unused device mask bits appear as masked.

READY-BUSY MODE REGISTER (INTEL)

The READY-BUSY MODE REGISTER (Attribute Memory Plane Address 4140H, Figure 12) provides the selection of two types of system interfacing for the busy-to-ready transition of the card's RDY/BSY# pin:

1. The standard PCMCIA READY-BUSY MODE, in which the card's RDY/BSY# signal generates a low-to-high transition (from busy to ready) only after *all* busy devices (not including masked devices) have completed their data-write or block-erase operations. This may result in a long interrupt latency.
2. A High-Performance mode that generates a low-to-high (from busy-to-ready) transition after each device becomes ready. This provides the host system with immediate notification that a specific device's operation has completed and that device may now be used. This is particularly useful in a file management application where a block pair, containing only deleted files, is being erased to free up space so new file data may be written.

Enabling the HIGH-PERFORMANCE READY-BUSY MODE requires a three step sequence:

1. Set all bits in the READY/BUSY MASK REGISTER. This prevents ready devices from triggering an unwanted interrupt when step 3 is performed.
2. Write 01H to the READY-BUSY MODE REGISTER. This sets the MODE bit.
3. Write 01H to the READY-BUSY MODE REGISTER. This clears the RACK bit.

The MODE and RACK bits *must* be written in the prescribed sequence, *not* simultaneously. The card's circuitry is designed purposely in this manner to prevent an initial, unwanted busy-to-ready transition. Note that in Step 2, writing to the RACK bit is a Don't Care.

When the High-Performance Mode is enabled, specific READY-BUSY MASK bits must be cleared after an operation is initiated on the respective devices. After each device becomes ready, the RDY/BSY# pin makes a low-to-high transition. To catch the next device's completion of an operation, the RACK bit must be cleared by writing "01H" to the Ready/Busy Mode Register.

READY-BUSY MODE REGISTER

(Read/Write Register)

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4140H	RESERVED FOR FUTURE USE						RACK	MODE

MODE = READY-BUSY MODE
0 = PCMCIA MODE
1 = HIGH PERFORMANCE

RACK = READY ACKNOWLEDGE CLEAR TO SET UP RDY/BSY# PIN, THEN CLEAR AFTER EACH DEVICE BECOMES READY TO ACKNOWLEDGE TRANSITION.

Figure 12. High Performance Ready-Busy Mode REGISTER (Intel) Used to Trigger a Ready Interrupt for Each Device

PRINCIPLES OF DEVICE OPERATION

Individual 28F008SA devices include a Command User Interface (CUI) and a Write State Machine (WSM) to manage write and erase functions in each device block.

The CUI serves as the device's interface to the Card Control Logic by directing commands to the appropriate device circuitry (Table 4). It allows for fixed power supplies during block erasure and data writes. The CUI handles the WE# interface into the device data and address latches, as well as system software requests for status while the WSM is operating.

The CUI itself does not occupy an addressable memory location. The CUI provides a latch used to store the command and address and data information needed to execute the command. Erase Setup and Erase Confirm commands require both appropriate command data and an address within the block to be erased. The Data Write Setup command requires both appropriate command data and the address of the location to be written, while the Data Write command consists of the data to be written and the address of the location to be written.

The CUI initiates flash memory writing and erasing operations only when V_{PP} is at 12V. Depending on the application, the system designer may choose to make the V_{PP} power supply switchable (available when writes and erases are required) or hardwired to V_{PPH}. When $V_{PP} = V_{PPL}$, power savings are incurred and memory contents cannot be altered. The CUI architecture provides protection from unwanted write and erase operations even when high voltage is applied to V_{PP}. Additionally, all functions are disabled whenever V_{CC} is below the write lockout voltage V_{LKO}, or when the card's Deep-Sleep modes are enabled. The WSM automates the writing and erasure of blocks within a device. This on-chip state machine controls block erase and data-write, freeing the host processor for other tasks. After receiving the Erase Setup and Erase Confirm commands from the CUI, the WSM controls block-erase. Progress is monitored via the device's status register, the card's control logic, and the RDY/BSY# pin of the PCMCIA interface. Data-write is similarly controlled, after destination address and expected data are supplied.

Table 4. Device Command Set

28F008SA Command[1]	Bus Cycles Req'd	First Bus Ccyle				Second Bus Cycle			
		Operation	Addr[2]	Data		Operation	Addr[2]	Data	
				x8 Mode	x16 Mode			x8 Mode	x16 Mode
Read Array/Reset	1	Write	DA	FFH	FFFFH				
Intelligent Identifer	3	Write	DA	90H	9090H	Read	IA	IID[3]	IID[3]
Read Device Status Register	2	Write	DA	70H	7070H	Read	DA	SRD[4]	SRD[4]
Clear Device Status Register	1	Write	DA	50H	5050H				
Erase Setup/Erase Confirm	2	Write	BA	20H	2020H	Write	BA	D0H	D0D0H
Erase Suspend/ Erase Resume	2	Write	DA	B0H	B0B0H	Write	DA	D0H	D0D0H
Write Setup/Write	2	Write	WA	40H	4040H	Write	WA	WD[5]	WD[5]
Alternate Write Setup/Write[6]	2	Write	WA	10H	1010H	Write	WA	WD[5]	WD[5]

NOTES:
1. Commands other than those shown above are reserved by Intel for future device implementations and should not be used.
2. DA = A device-level (or device pair) address within the card.
   BA = Address within the block of a specific device (device pair) being erased.
   WA = Address of memory location to be written.
   IA = A device-level address; 00H for manufacturer code, 01 for device code.
3. Following the intelligent identifier command, two read operations access manufacturer (89H) and device codes (A2H).
4. SRD = Data read from Device Status Register.
5. WD = Data to be written at location WA. Data is latched on the rising edge of WE#.
6. Either 40H or 10H are recognized by the WSM as the Write Setup command.

6

COMMAND DEFINITIONS

Read Array (FFH) —

Upon initial card power-up, after exit from the Deep-Sleep modes, and whenever illegal commands are given, individual devices default to the Read Array mode. This mode is also entered by writing FFH into the CUI. In this mode, microprocessor read cycles retrieve array data. Devices remain enabled for reads until the CUI receives an alternate command. Once the internal WSM has started a block-erase or data-write operation within a device, that device will not recognize the Read Array command until the WSM has completed its operation (or the Erase Suspend command is issued during erase).

Intelligent Identifier (90H) —

After executing this command, the intelligent identifier values can be read. Only address A_0 of each device is used in this mode, all other address inputs are ignored [(Manufacturer code = 89H for A_0 = 0), (Device code = A2H for A_0 = 1)]. The device will remain in this mode until the CUI receives another command.

This information is useful by system software in determining what type of flash memory device is contained within the card and allows the correct matching of device to write and erase algorithms. System software that fully utilizes the PCMCIA specification will not use the intelligent identifier mode, as this data is available within the Card Information Structure (refer to section on PCMCIA Card Information Structure).

Read Status Register (70H)

After writing this command, a device read outputs the contents of its Status Register, regardless of the address presented to that device. The contents of this register are latched on the falling edge of OE#, CE_1# (and/or CE_2#), whichever occurs last in the read cycle. This prevents possible bus errors which might occur if the contents of the Status Register changed while reading its contents. CE_1# (and CE_2# for odd-byte or word access) or OE# must be toggled with each subsequent status read, or the completion of a write or erase operation will not be evident. This command is executable while the WSM is operating, however, during a block-erase or data-write operation, reads from the device will automatically return status register data. Upon completion of that operation, the device remains in the Status Register read mode until the CUI receives another command.

The read Status Register command functions when $V_{PP} = V_{PPL}$ or V_{PPH}.

Clear Status Register (50H)

The Erase Status and Write Status bits may be set to "1"s by the WSM and can only be reset by the Clear Status Register Command. These bits indicate various failure conditions. By allowing system software to control the resetting of these bits, several operations may be performed (such as cumulatively writing several bytes or erasing multiple blocks in sequence). The device's Status Register may then be polled to determine if an error occurred during that sequence. This adds flexibility to the way the device may be used.

Additionally, the V_{PP} Status bit (SR.3) MUST be reset by system software (Clear Status Register command) before further block-erases are attempted (after an error).

The Clear Status Register command functions when $V_{PP} = V_{PPL}$ or V_{PPH}. This command puts the device in the Read Array mode.

Write Setup/Write

A two-command sequence executes a data-write operation. After the system switches V_{PP} to V_{PPH}, the write setup command (40H) is written to the CUI of the appropriate device, followed by a second write specifying the address and write data (latched on the rising edge of WE#). The device's WSM controls the data-write and write verify algorithms internally. After receiving the two-command write sequence, the device automatically outputs Status Register data when read (see Figure 13). The CPU detects the completion of the write operation by analyzing card-level or device-level indicators. Card-level indicators include the RDY/BSY# pin and the READY-BUSY STATUS REGISTER; while device-level indicators include the specific device's Status Register. Only the Read Status Register command is valid while the write operation is active. Upon completion of the data-write sequence (see section on Status Register) the device's Status Register reflects the result of the write operation. The device remains in the Read Status Register mode until the CUI receives an alternate command.

Erase Setup/Erase Confirm Commands (20H)

Within a device, a two-command sequence initiates an erase operation on one device block at a time. After the system switches V_{PP} to V_{PPH}, an Erase Setup command (20H) prepares the CUI for the Erase Confirm command (D0H). The device's WSM controls the erase algorithms internally. After receiving the two-command erase sequence, the device automatically outputs Status Register data when read (see Figure 14). If the command after erase setup is not an Erase Confirm command, the CR sets the Write Failure and Erase Failure bits of the Status Register, places the device into the Read Status Register mode, and waits for another command. The Erase Confirm command enables the WSM for erase (simultaneously closing the address latches for that device's block (A_{16}–A_{19}). The CPU detects the completion of the erase operation by analyzing card-level or device-level indicators. Card-level indicators include the RDY/BSY pin and the READY-BUSY STATUS REGISTER; while device-level indicators include the specific device's Status Register. Only the Read Status Register and Erase Suspend command is valid during an active erase operation. Upon completion of the erase sequence (see section on Status Register) the device's Status Register reflects the result of the erase operation. The device remains in the Read Status Register mode until the CUI receives an alternate command.

The two-step block-erase sequence ensures that memory contents are not accidentally erased. Erase attempts while $V_{PPL} < V_{PP} < V_{PPH}$ produce spurious results and are not recommended. Reliable block erasure only occurs when $V_{PP} = V_{PPH}$. In the absence of this voltage, memory contents are protected against erasure. If block erase is attempted while $V_{PP} = V_{PPL}$, the V_{PP} Status bit will be set to "1".

When erase completes, the Erase Status bit should be checked. If an erase error is detected, the device's Status Register should be cleared. The CUI remains in Read Status Register mode until receiving an alternate command.

Erase Suspend (B0H)/Erase Resume (D0H)

Erase Suspend allows block erase interruption to read data from another block of the device or to temporarily conserve power for another system operation. Once the erase process starts, writing the Erase Suspend command to the CUI (see Figure 15) requests the WSM to suspend the erase sequence at a predetermined point in the erase algorithm. In the erase suspend state, the device continues to output Status Register data when read.

Polling the device's RY/BY# and Erase Suspend Status bits (Status Register) will determine when the erase suspend mode is valid. It is important to note that the card's RDY/BSY# pin will also transition to V_{OH} and will generate an interrupt if this pin is connected to a system-level interrupt. At this point, a Read Array command can be written to the device's CUI to read data from blocks **other than those which are suspended**. The only other valid commands at this time are Read Status Register (70H) and Erase Resume (D0H). If V_{PP} goes low during Erase Suspend, the V_{PP} Status bit is set in the Status Register and the erase operation is aborted.

The Erase Resume command clears the Erase Suspend state and allows the WSM to continue with the erase operation. The device's RY/BY# Status and Erase Suspend Status bits and the card's READY-BUSY Status Register are automatically updated to reflect the erase resume condition. The card's RDY/BSY# pin also returns to V_{OL}.

Invalid/Reserved

These are unassigned commands having the same effect as the Read Array command. Do not issue any command other than the valid commands specified above. Intel reserves the right to redefine these codes for future functions.

6

DEVICE STATUS REGISTER

Each 28F008SA device in the Series 2 Card contains a Status Register which displays the condition of its Write State Machine. The Status Register is read at any time by writing the Read Status command to the CUI. After writing this command, all subsequent Read operations output data from the Status Register, until another command is written to the CUI.

Bit 7—WSM Status

This bit reflects the Ready/Busy condition of the WSM. A "1" indicates that read, block-erase or data- write operations are available. A "0" indicates that write or erase operations are in progress.

Bit 6—Erase Suspend Status

If an Erase Suspend command is issued during the erase operation, the WSM halts execution and sets the WSM Status bit and the Erase Suspend Status bit to a "1". This bit remains set until the device receives an Erase Resume command, at which point the CUI resets the WSM Status bit and the Erase Suspend Status bit.

Bit 5—Erase Status

This bit will be cleared to 0 to indicate a successful block-erasure. When set to a "1", the WSM has been unsuccessful at performing an erase verification. The device's CUI only resets this bit to a "0" in response to a Clear Status Register command.

Bit 4—Write Status

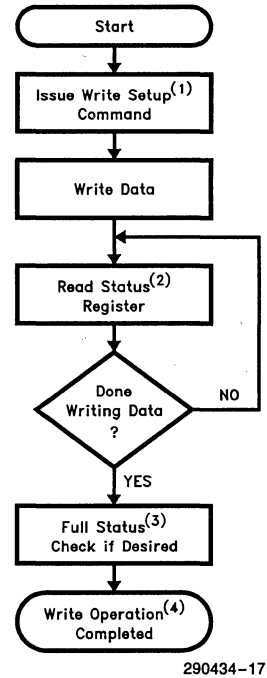
This bit will be cleared to a 0 to indicate a successful data-write operation. When the WSM fails to write data after receiving a write command, the bit is set to a "1" and can only be reset by the CUI in response to a Clear Status Register command.

Bit 3—V_{PP} Status

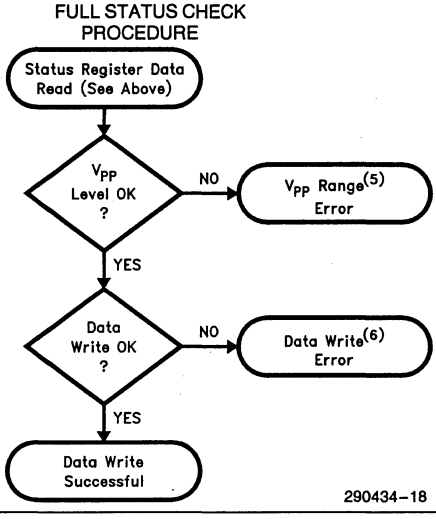
During block-erase and data-write operations, the WSM monitors the output of the device's internal V_{PP} detector. In the event of low V_{PP}, the WSM sets ("1") the V_{PP} Status bit, the status bit for the operation in progress (either write or erase). The CUI resets these bits in response to a Clear Status Register command. Also, the WSM RY/BY# bit will be set to indicate a device ready condition. This bit MUST be reset by system software (Clear Status Register command) before further data writes or block erases are attempted.

Device Status Register (Read Only Register)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WSM Status	Erase Suspend Status	Erase Status	Write Status	V_{PP} Status	Reserved		



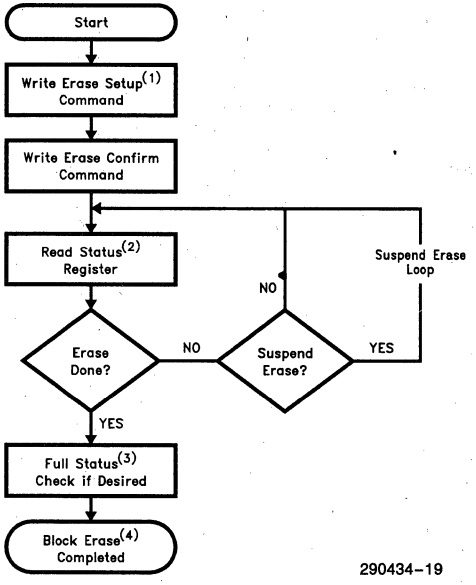
Bus Operation	Command	x8 Mode	x16 Mode
Write	Write Setup	Data = 40H Address = Byte Within Card to be Written	Data = 4040H Address = Word Within Card to be Written
Write	Data Write	Data to be Written Address = Byte Within Card to be Written	Data to be Written Address = Word Within Card to be Written
Read	Defaults to Device Status Register Read Mode	Status Register Data. Toggle OE#, CE_1# or CE_2# to update Status Register	Status Register Data. Toggle OE# or (CE_1# and CE_2#) to update Status Registers
Standby		Check SR Bit 7 1 = Ready, 0 = Busy	Check SR Bits 7 and 15 1 = Ready, 0 = Busy



Bus Operation	Command	x8 Mode	x16 Mode
Standby		Check SR Bit 3 1 = V_{PP} Detected Low	Check SR Bits 3 and 11 1 = V_{PP} Detected Low
Standby		Check SR Bit 4 1 = Data Write Error	Check SR Bits 4 and 12 1 = Data Write Error

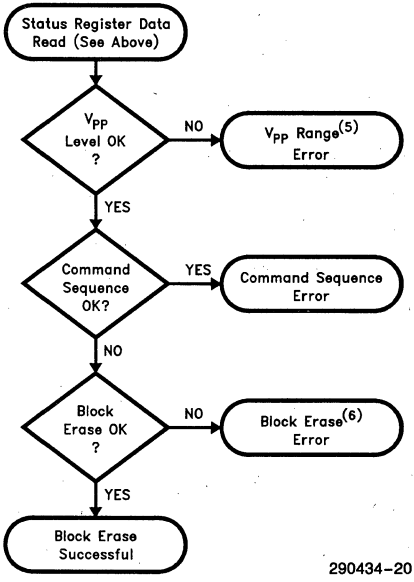
Figure 13. Device-Level Automated Write Algorithm

NOTES:
1. Repeat for subsequent data writes.
2. In addition, the card's READY-BUSY STATUS REGISTER or the RDY/BSY# pin may be used.
3. Full device-level status check can be done after each data write or after a sequence of data writes.
4. Write FFH (or FFFFH) after the last data write operation to reset the device(s) to Read Array Mode.
5. If a data write operation fails due to a low V_{PP} (setting SR Bit 3), the Clear Status Register command MUST be issued before further attempts are allowed by the Write State Machine.
6. If a data write operation fails during a multiple write sequence, SR Bit 4 (Write Status) will not be cleared until the Command User Interface receives the Clear Status Register command.



Bus Operation	Command	x8 Mode	x16 Mode
Write	Erase Setup	Data = 20H Address = Block Within Card to be Erased	Data = 2020H Address = Block Pair Within Card to be Erased
Write	Erase	Data = D0H Address = Block Within Card to be Erased	Data = D0D0H Address = Block Pair Within Card to be Erased
Read	Defaults to Device Status Register Read Mode	Status Register Data. Toggle OE#, CE_1# or CE_2# to update Status Register	Status Register Data. Toggle OE# or (CE_1# and CE_2#) to update Status Register
Standby		Check SR Bit 7 1 = Ready, 0 = Busy	Check SR Bits 7 and 15 1 = Ready, 0 = Busy

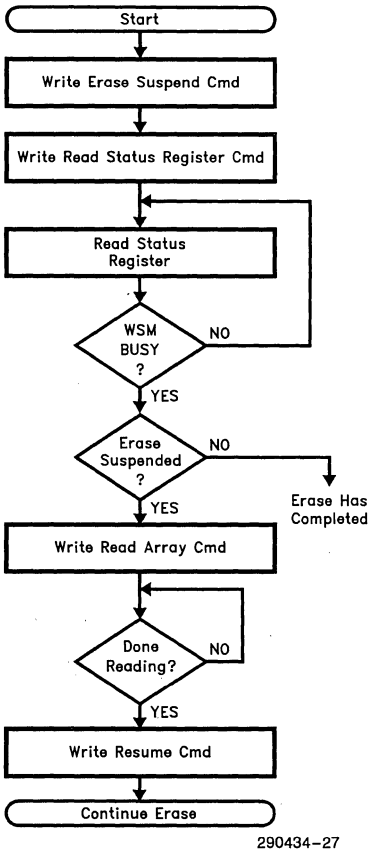
FULL STATUS CHECK PROCEDURE



Bus Operation	Command	x8 Mode	x16 Mode
Standby		Check SR Bit 3 1 = V_{PP} Detected Low	Check SR Bits 3 and 11 Either Bit 1 = V_{PP} Detected Low
Standby		Check SR Bits 4 and 5 Both 1 = Command Sequence Error	Check SR Bits 4, 5, 12, 13 All 1 = Command Sequence Error
Standby		Check SR Bit 5 1 = Block Erase Error	Check SR Bits 5 and 13 Both 1 = Block Erase Failure

Figure 14. Device-Level Automated Erase Algorithm

NOTES:
1. Repeat for subsequent data writes.
2. In addition, the card's READY-BUSY STATUS REGISTER or the RDY/BSY# pin may be used.
3. Full device-level status check can be done after each block erase or after a sequence of block erases.
4. Write FFH (or FFFFH) after the last block erase operation to reset the device(s) to Ready Array Mode.
5. If a block erase operation fails due to a low V_{PP} (setting SR Bit 3), the Clear Status Register command MUST be issued before further attempts are allowed by the Write State Machine.
6. If a block erase operation fails during a multiple block erase sequence, SR Bit 4 (Write Status) will not be cleared until the Command User Interface receives the Clear Status Register command.



Bus Operation	Command	x8 Mode	x16 Mode
Write	Suspend Erase	Data = B0H, Address = Desired Block to Erase Suspend	Data = B0B0H, Address = Desired Block Pair to Erase Suspend
Read		Status Register Data. Toggle OE#, CE_1# or CE_2# to update Status Register	Status Register Data. Toggle OE# or (CE_1# and CE_2#) to update Status Register
Standby		Check SR Bit 7 1 = Ready, 0 = Busy	Check SR Bit 7 and 15 1 = Ready, 0 = Busy
Standby		Check SR Bit 6 1 = Suspended, 0 = In Progress	Check SR Bit 6 and 14 1 = Suspended, 0 = In Progress
Write	Rd Array Cmd	Data = FFH	Data = FFFFH
Read		Read Data until finished	Read Data until finished
Write	Erase Resume	Data = D0H, Address = Valid Block Address.	Data = D0D0H, Address = Valid Block Pair Address.

Figure 15. Erase Suspend/Resume Algorithm. Allows Reads to Interrupt Erases.

6

POWER CONSUMPTION

STANDBY MODE

In most applications, software will only be accessing one device pair at a time. The Series 2 Card is defined to be in the standby mode when one device pair is in the Read Array Mode while the remaining devices are in the Deep-Sleep Mode. The Series 2 Card's CE_1# and CE_2# input signals must also be at V_{IH}. In standby mode, much of the card's circuitry is shut off, substantially reducing power consumption. Typical power consumption for a 20 Megabyte Series 2 card in standby mode is 65 μA.

SLEEP MODE

Writing a "1" to the PWRDWN bit of the GLOBAL POWERDOWN REGISTER places all FlashFile Memory devices into a Deep-Sleep mode. This disables most of the 28F008SA's circuitry and reduces current consumption to 0.2 μA per device. Additionally, when the host system pulls ASIC control logic high and latches all address and data lines (i.e., not toggling), the card's total current draw is reduced to approximately 5 μA (CMOS input levels) for a 20 Megabyte card. On writing a "0" to the PWRDWN bit (Global PowerDown Register) or any individual device pair (Sleep Control Register), a Deep-Sleep mode recovery period must be allowed for 28F008SA device circuitry to power back on.

SYSTEM DESIGN CONSIDERATIONS

POWER SUPPLY DECOUPLING

Flash memory power-switching characteristics require careful device decoupling. System designers are interested in three supply current issues—standby, active and transient current peaks, produced by rising and falling edges of CE_1# and CE_2#. The capacitive and inductive loads on the card and internal flash memory device pairs determine the magnitudes of these peaks.

The Flash Memory Card features on-card ceramic decoupling capacitors connected between V_{CC} and GND, and between V_{PP1}/V_{PP2} and GND to help transient voltage peaks.

On the host side, the card connector should also have a 4.7 μF electrolytic capacitor between V_{CC} and GND, as well as between V_{PP1}/V_{PP2} and GND. The bulk capacitors will overcome voltage slumps caused by printed-circuit-board trace inductance, and will supply charge to the smaller capacitors as needed.

POWER UP/DOWN PROTECTION

Each device in the Flash Memory Card is designed to offer protection against accidental erasure or writing, caused by spurious system-level signals that may exist during power transitions. The card will power-up into the Read Array Mode.

A system designer must guard against active writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and CE_1# (and/or CE_2#) must be low for a command write, driving either to V_{IH} will inhibit writes. With its Command User Interface, alteration of device contents only occurs after successful completion of the two-step command sequences.

While these precautions are sufficient for most applications, an alternative approach would allow V_{CC} to reach its steady state value before raising V_{PP1}/V_{PP2} above V_{CC} + 2.0V. In addition, upon powering-down, V_{PP1}/V_{PP2} should be below V_{CC} + 2.0V, before lowering V_{CC}.

HOT INSERTION/REMOVAL

The capability to remove or insert PC cards while the system is powered on (i.e., hot insertion/removal) requires careful design approaches on the system and card levels. To design for this capability consider card overvoltage stress, system power droop and control line stability.

A PCMCIA/JEIDA specified socket properly sequences the power supplies to the flash memory card via shorter and longer pins. This assures that hot insertion and removal will not result in card damage or data loss.

PCMCIA CARD INFORMATION STRUCTURE

The Card Information Structure (CIS) starts at address zero of the card's Attribute Memory Plane. It contains a variable-length chain of data blocks (tuples) that conform to a basic format as shown in Table 5. This section describes each tuple contained within the Series 2 Flash Memory Card.

The Device Information Tuple

This tuple (CISTPL_DEV = 01H) contains information pertaining to the card's speed and size. The Series 2 Card is offered with a 200 nanosecond access time. Card sizes range between 2 and 20 Megabytes.

Table 5. Tuple Format

Bytes	Data
0	Tuple Code: CISTPL_xxx. The tuple code 0FFH indicates no more tuples in the list.
1	Tuple Link: TPL_LINK. Link to the next tuple in the list. This can be viewed as the number of additional bytes in tuple, excluding this byte. If the link field is zero, the tuple body is empty. If the link field contains 0FFH, this tuple is the last tuple in the list.
2–n	Bytes specific to this tuple.

The Device Geometry Tuple

This tuple (CISTPL_DEVICEGEO = 1EH) is conceptually similar to a DOS disk geometry tuple (CISTPL_GEOMETRY), except it is not a format-dependent property; this deals with the fixed architecture of the memory device(s).

Fields are defined as follows:

DGTPL BUS—Value = n, where system bus width = $2^{(n-1)}$ bytes. N = 2 for standard PCMCIA Release 1.0/2.0 cards.

DGTPL EBS—Value = n, where the memory array's physical memory segments have a minimum erase block size of $2^{(n-1)}$ address increments of DGTPL_BUS-wide accesses.

DGTPL RBS—Value = n, where the memory array's physical memory segments have a minimum read block size of $2^{(n-1)}$ address increments of DGTPL_BUS-wide accesses.

DGTPL WBS—Value = n, where the memory array's physical memory segments have a minimum write block size of $2^{(n-1)}$ address increments of DGTPL_BUS-wide accesses.

DGTPL PART—Value = n, where the memory array's physical memory segments can have partitions subdividing the arrays in minimum granularity of $2^{(n-1)}$ number of erase blocks.

FL DEVICE INTERLEAVE—Value = n, where card architectures employ a multiple of $2^{(n-1)}$ times interleaving of the entire memory arrays with the above characteristics. Non-interleaved cards have values n = 1.

Jedec Programming Information Tuple

This tuple (CISTPL_JEDEC = 18H) contains the Intel manufacturing identifier (89H) and the 28F008SA device ID (A2H).

Level 1 Version/Product Information Tuple

This tuple (CISTPL_VERI = 15H) contains Level-1-version compliance and card-manufacturer information. Fields are described as follows:

TPLLV1 MAJOR—Major version number = 04H.

TPLLV1 MINOR—Minor version number = 01H for release 2.0.

TPLLV1 INFO—

Name of manufacturer	= intel;
Name of product	= SERIES2-"Card size";
Card type	= 2;
Speed	= 150 ns or 200 ns
Register Base	= REGBASE 4000H
Test Codes	= DBBDRELP
Legalities	= COPYRIGHT intel Corporation 1991

The Configurable Card Tuple

This tuple (CISTPL_CONF = 1AH) describes the interface supported by the card and the locations of the Card Configuration Registers and the Card Configuration Table.

Fields are described as follows:

TPCC SZ—Size of fields byte = 01H.

TPCC LAST—Index number of the last entry in the Card Configuration Table = 00H.

TPCC RADR—Configuration Registers Base Address in Reg Space = 4000H.

TPCC RMSK—Configuration Registers Present Mask = 03H.

The End-Of-List Tuple

The end-of-list tuple (CISTPL_END = FFH) marks the end of a tuple chain. Upon encountering this tuple, continue tuple processing as if a long-link to address 0 of common memory space were encountered.

Tuple Address	Value	Description
00H	01H	CISTPL_DEV
02H	03H	TPL_LINK
04H	53H 52H	DEVICE_INFO = FLASH 150 ns DEVICE_INFO = FLASH 200 ns
06H	06H 0EH 26H 4EH	CARD SIZE 2M 4M 10M 20M
08H	FFH	END OF DEVICE
0AH	1EH	CISTPL_ DEVICEGEO
0CH	06H	TPL_LINK
0EH	02H	DGTPL_BUS
10H	11H	DGTPL_EBS
12H	01H	DGTPL_RBS
14H	01H	DGPL_WBS
16H	03H	DGTPL_PART
18H	01H	FL_DEVICE INTERLEAVE
1AH	18H	CISTPL_JEDEC
1CH	02H	TPL_LINK
1EH	89H	INTEL J-ID
20H	A2H	28F008 J-ID
22H	15H	CISTPL_VER1
24H	50H	TPL_LINK
26H	04H	TPLLV1 MAJOR
28H	01H	TPLLV1 MINOR
2AH	69H	TPLLV1 INFO i
2CH	6EH	n
2EH	74H	t
30H	65H	e

Tuple Address	Value	Description
32H	6CH	l
34H	00H	END TEXT
36H	53H	S
38H	45H	E
3AH	52H	R
3CH	49H	I
3EH	45H	E
40H	53H	S
42H	32H	2
44H	2DH	—
46H	30H 30H 31H 32H	2M = 0 4M = 0 10M = 1 20M = 2
48H	32H 34H 30H 30H	2M = 2 4M = 4 10M = 0 20M = 0
4AH	20H	SPACE
4CH	00H	END TEXT
4EH	32H	CARD TYPE 2
50H	41H 42H 45H 5AH 48H 49H 4CH 4FH	A = 2M, 150 ns B = 4M, 150 ns E = 10M, 150 ns Z = 20M, 150 ns H = 2M, 200 ns I = 4M, 200 ns L = 10M, 200 ns O = 20M, 200 ns

Tuple Address	Value	Description
52H	20H	SPACE
54H	52H	REGBASE-R
56H	45H	E
58H	47H	G
5AH	42H	B
5CH	41H	A
5EH	53H	S
60H	45H	E
62H	20H	SPACE
64H	34H	4000h 4
66H	30H	0
68H	30H	0
6AH	30H	0
6CH	68H	h
6EH	20H	SPACE
70H	44H	D
72H	42H	B
74H	42H	B
76H	44H	D
78H	52H	R
7AH	45H	E
7CH	4CH	L
7EH	50H	P
80H	00H	END TEXT
82H	43H	COPYRIGHT C
84H	4FH	O
86H	50H	P
88H	59H	Y
8AH	52H	R
8CH	49H	I
8EH	47H	G
90H	48H	H
92H	54H	T
94H	20H	SPACE

Tuple Address	Value	Description
96H	69H	i
98H	6EH	n
9AH	74H	t
9CH	65H	e
9EH	6CH	l
A_0H	20H	SPACE
A2H	43H	CORPORATION C
A4H	4FH	O
A6H	52H	R
A8H	50H	P
AAH	4FH	O
ACH	52H	R
AEH	41H	A
B0H	54H	T
B2H	49H	I
B4H	4FH	O
B6H	4EH	N
B8H	20H	SPACE
BAH	31H	1
BCH	39H	9
BEH	39H	9
C0H	31H	1
C2H	00H	END TEXT
C4H	FFH	END OF LIST
C6H	1AH	CISTPL__CONF
C8H	06H	TPL__LINK
CAH	01H	TPCC__SZ
CCH	00H	TPCC__LAST
CEH	00H	TPCC__RADR
D0H	40H	TPCC__RADR
D2H	03H	TPCC__RMSK
D4H	FFH	END OF LIST
D6H	FFH	CISTPL__END
D8H	00H	INVALID ECIS ADDRESS

OPERATING SPECIFICATIONS

NOTICE: This data sheet contains preliminary information on new products in production. The specifications are subject to change without notice. Verify with your local Intel Sales office that you have the latest data sheet before finalizing a design.

ABSOLUTE MAXIMUM RATINGS*

Commercial Temperature
- Operating Range ..... 0°C to +70°C(1)
- Storage Temperature ..... −40°C to +85°C

Extended Temperature
- Operating Range ..... −40°C to +85°C
- Lower Temperature Limit ..... −40°C
- Upper Temperature Limit ..... +85°C

Voltage on Any Pin with
Respect to Ground ..... −2.0V to V_{CC} + 2.0V(2)

V_{PP1}/V_{PP2} Supply Voltage with
Respect to Ground .. −2.0V to V_{CC} + 14.0V(2, 3)

V_{CC} Supply Voltage with
Respect to Ground ..... −0.5V to +7.0V

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for commercial product defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to −2.0V tor periods less than 20 ns. Maximum DC voltage on output pins is V_{CC} + 0.5V, which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.
3. Maximum DC input voltage on V_{PP1}/V_{PP2} may overshoot to +14.0V tor periods less than 20 ns.
4. V_{PP} generator turned "on" for 3.3V or 5.0V only operation.

OPERATING CONDITIONS

Symbol	Parameter	Min	Max	Unit
T_A	Operating Temperature	0	70	°C
V_{CC}	V_{CC} Supply Voltage (5%)	4.75	5.25	V

COMMON DC CHARACTERISTICS, CMOS and TTL

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Condition
I_{LI}	Input Leakage Current	1, 3		±1	±20	µA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or GND
I_{LO}	Output Leakage Current	1		±1	±20	µA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or GND
V_{IL}	Input Low Voltage	1	−0.5		0.8	V	
V_{IH}	Input High Voltage (TTL)	1	2.4		V_{CC} + 0.3	V	
	Input High Voltage (CMOS)		0.7 V_{CC}		V_{CC} + 0.3		
V_{OL}	Output Low Voltage	1	V_{SS}		0.4	V	$V_{CC} = V_{CC}$ Min I_{OL} = 3.2 mA
V_{OH}	Output High Voltage	1	4.0		V_{CC}	V	$V_{CC} = V_{CC}$ Min I_{OH} = −2.0 mA
V_{PPL}	V_{PP} during Read Only Operations	1, 2	0.0		6.5	V	
V_{PPH}	V_{PP} during Read/Write Operations	1	11.4		12.6	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage	1	2.0			V	

NOTES:
1. Values are the same for byte and word wide modes and for all card densities.
2. Block Erases/Data Writes are inhibited when V_{PP} and V_{PPL} and not guaranteed in the range between V_{PPH} and V_{PPL}.
3. Exceptions: With V_{IN} = GND, the leakage on CE_1#, CE_2#, REG#, OE#, WE#, will be ≤ 500 µA due to internal pullup resistors and, with $V_{IN} = V_{CC}$, RST leakage will be ≤ 500 µA due to internal pulldown resistor.

DC CHARACTERISTICS, CMOS

Symbol	Parameter		Notes	Byte Wide Mode			Word Wide Mode			Unit	Test Condition
				Min	Typ	Max	Min	Typ	Max		
I_{CCR}	V_{CC} Read Current		1, 3		45	85		65	120	mA	$V_{CC} = V_{CC}$ Max, Control Signals = GND t_{CYCLE} = 200 ns, I_{OUT} = 0 mA
I_{CCW}	V_{CC} Write Current		1, 3		35	80		45	110	mA	Data Write in Progress
I_{CCE}	V_{CC} Erase Current		1, 2, 3		35	80		45	110	mA	Block (Pair) Erase in Progress
I_{CCS}	V_{CC} Standby Current	2 Meg	1, 4, 6		61	220		61	220	µA	$V_{CC} = V_{CC}$ Max, Control Signals = V_{IH}
		4 Meg			62	222		62	222		
		10 Meg			63	230		63	230		
		20 Meg			65	242		65	242		
I_{CCSL}	V_{CC} Sleep Current	2 Meg	1, 4, 5		1	22		1	22	µA	
		4 Meg			2	25		2	25		
		10 Meg			3	32		3	32		
		20 Meg			5	44		5	44		
I_{PPW}	V_{PP} Write Current ($V_{PP} = V_{PPH}$)		1, 3		10	30		20	60	mA	Data Write in Progress
I_{PPE}	V_{PP} Erase Current ($V_{PP} = V_{PPH}$)		1, 3		10	30		20	60	mA	Block (Pair) Erase in Progress
I_{PPSL}	V_{PP} Sleep Current	2 Meg	1, 5		0.2	10		0.2	10	µA	
		4 Meg			0.4	20		0.4	20		
		10 Meg			1	50		1	50		
		20 Meg			2	100		2	100		
I_{PPS1}	V_{PP} Standby or Read Current ($V_{PP} \leq V_{CC}$)	2 Meg	1, 6		2.0	20		2.0	20	µA	
		4 Meg			2.2	30		2.2	30		
		10 Meg			2.8	60		2.8	60		
		20 Meg			3.8	110		3.8	110		
I_{PPS2}	V_{PP} Standby or Read Current ($V_{PP} = V_{CC}$)	2 Meg	1, 6		180	400		180	400	µA	
		4 Meg			180	410		180	410		
		10 Meg			181	440		181	440		
		20 Meg			182	490		182	490		

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. The Data Sheet specification for the 28F008SA in Erase Suspend (I_{CCES}) is 5 mA typical and 10 mA max with the device deselected. If the device(s) are read while in Erase Suspend Mode, current draw is the sum of I_{CCES} and I_{CCR}.
3. Standby or Sleep currents are not included for non-accessed devices.
4. Address and data inputs to card static. Control line voltages equal to V_{IH} or V_{IL}.
5. All 28F008SA devices in Deep-Sleep (Reset-PowerDown) mode.
6. In Byte and Word Mode, all but two devices in Deep-Sleep.

DC CHARACTERISTICS, TTL

Symbol	Parameter		Notes	Byte Wide Mode			Word Wide Mode			Unit	Test Condition
				Min	Typ	Max	Min	Typ	Max		
I_{CCR}	V_{CC} Read Current		1, 3		70	135		90	170	mA	$V_{CC} = V_{CC}$ Max, Control Signals = GND t_{CYCLE} = 200 ns, I_{OUT} = 0 mA
I_{CCW}	V_{CC} Write Current		1, 3		60	130		70	160	mA	Data Write in Progress
I_{CCE}	V_{CC} Erase Current		1, 2, 3		60	130		70	160	mA	Block (Pair) Erase in Progress
I_{CCS}	V_{CC} Standby Current	2 Meg	1, 4, 6		20	100		20	100	mA	$V_{CC} = V_{CC}$ Max, Control Signals = V_{IH}
		4 Meg									
		10 Meg									
		20 Meg									
I_{CCSL}	V_{CC} Sleep Current	2 Meg	1, 4, 5		20	100		20	100	mA	
		4 Meg									
		10 Meg									
		20 Meg									
I_{PPW}	V_{PP} Write Current ($V_{PP} = V_{PPH}$)		1, 3		10	30		20	60	mA	Data Write in Progress
I_{PPE}	V_{PP} Erase Current ($V_{PP} = V_{PPH}$)		1, 3		10	30		20	60	mA	Block (Pair) Erase in Progress
I_{PPSL}	V_{PP} Sleep Current	2 Meg	1, 5		0.2	10		0.2	10	µA	
		4 Meg			0.4	20		0.4	20		
		10 Meg			1.0	50		1.0	50		
		20 Meg			2.0	100		2.0	100		
I_{PPS1}	V_{PP} Standby or Read Current ($V_{PP} \leq V_{CC}$)	2 Meg	1, 6		2.0	20		2.0	20	µA	
		4 Meg			2.2	30		2.2	30		
		10 Meg			2.8	60		2.8	60		
		20 Meg			3.8	110		3.8	110		
I_{PPS2}	V_{PP} Standby or Read Current ($V_{PP} = V_{CC}$)	2 Meg	1, 6		180	400		180	400	µA	
		4 Meg			180	410		180	410		
		10 Meg			181	440		181	440		
		20 Meg			182	490		182	490		

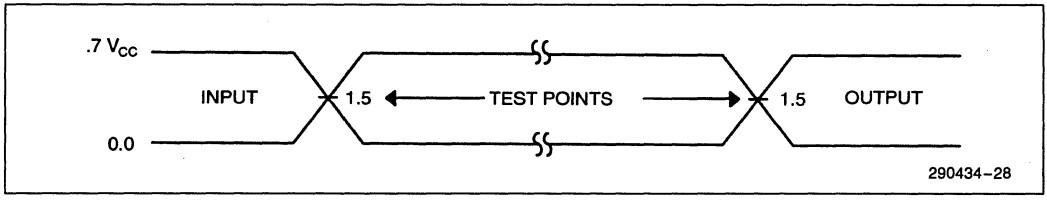
NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. The Data Sheet specification for the 28F008SA in Erase Suspend (I_{CCES}) is 5 mA typical and 10 mA max with the device deselected. If the device(s) are read while in Erase Suspend Mode, current draw is the sum of I_{CCES} and I_{CCR}.
3. Standby or Sleep currents are not included for non-accessed devices.
4. Address and data inputs to card static. Control line voltages equal to V_{IH} or V_{IL}.
5. All 28F008SA devices in Deep-Sleep (Reset-PowerDown) mode.
6. In Byte and Word Mode, all but two devices in Deep-Sleep.
7. The current consumption from the 28F008SA is insignificant in relation to the ASIC's.

AC CHARACTERISTICS

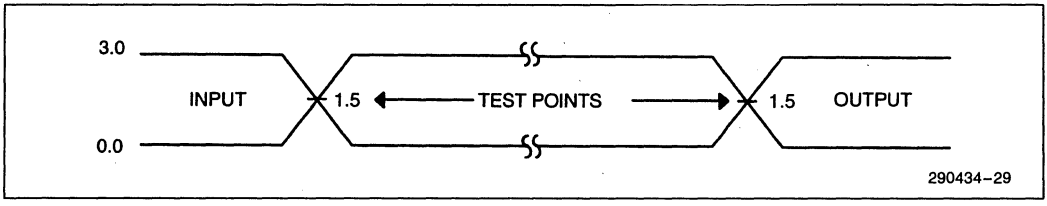
AC Timing Diagrams and characteristics are guaranteed to meet or exceed PCMCIA Release 2.0 specifications. PCMCIA allows a 300 ns access time for Attribute Memory. Note that read and write access timings to the Series 2 Flash Memory Card's Common and Attribute Memory Planes are identical at 200 ns. Furthermore, there is no delay in switching between the Common and Attribute Memory Planes.

COMMON AND ATTRIBUTE MEMORY, AC CHARACTERISTICS: Read-Only Operations

Symbol		Parameter	Notes	Min	Max	Unit
JEDEC	PCMCIA					
t_{AVAV}	t_{RC}	Read Cycle Time		200		ns
t_{AVQV}	t_a (A)	Address Access Time			200	ns
t_{ELQV}	t_a (CE)	Card Enable Access Time			200	ns
t_{GLQV}	t_a (OE)	Output Enable Access Time			100	ns
t_{EHQX}	t_{dis} (CE)	Output Disable Time from CE#			90	ns
t_{GHQZ}	t_{dis} (CE)	Output Disable Time from OE#			70	ns
t_{GLQX}	t_{en} (CE)	Output Enable Time from CE#		5		ns
t_{ELQX}	t_{en} (OE)	Output Enable Time from OE#		5		ns
t_{AXQX}	t_v (A)	Data Valid from Add Change		0		ns
t_{RHQV}		Reset-PwrDwn Recovery to Output Delay		500		ns
	t_{su} (V_{CC})	CE Setup Time on Power-Up		1		ms
		First Access after Reset		500		ns

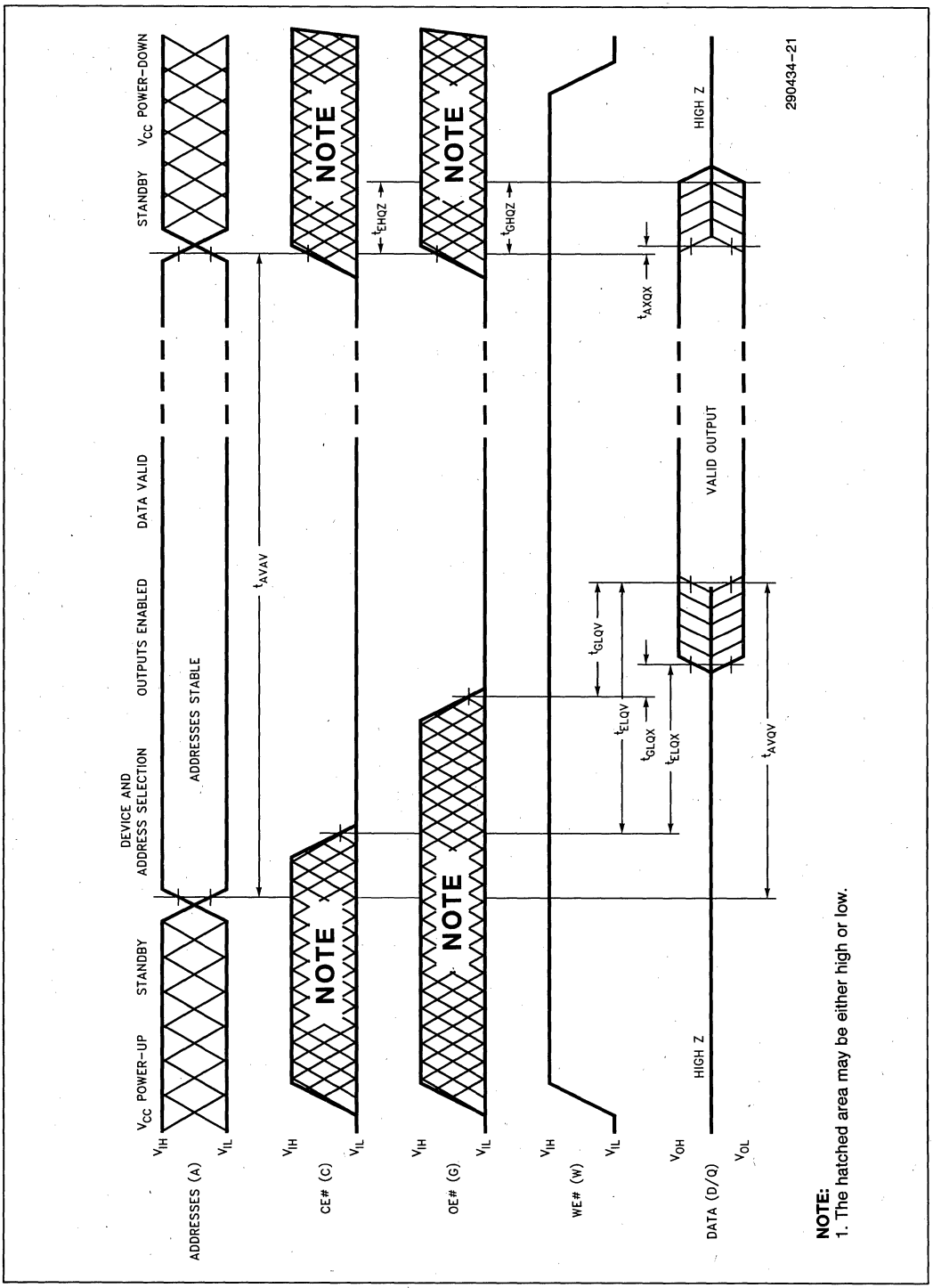


Transient Input/Output Reference Waveform (V_{CC} = 5.0V) for Standard Test Configuration



Transient Input/Output Reference Waveform (V_{CC} = 3.3V) for Standard Test Configuration

6



NOTE:
1. The hatched area may be either high or low.

Figure 16. AC Waveform for Read Operations

COMMON AND ATTRIBUTE MEMORY, AC CHARACTERISTICS: Write Operations[1]

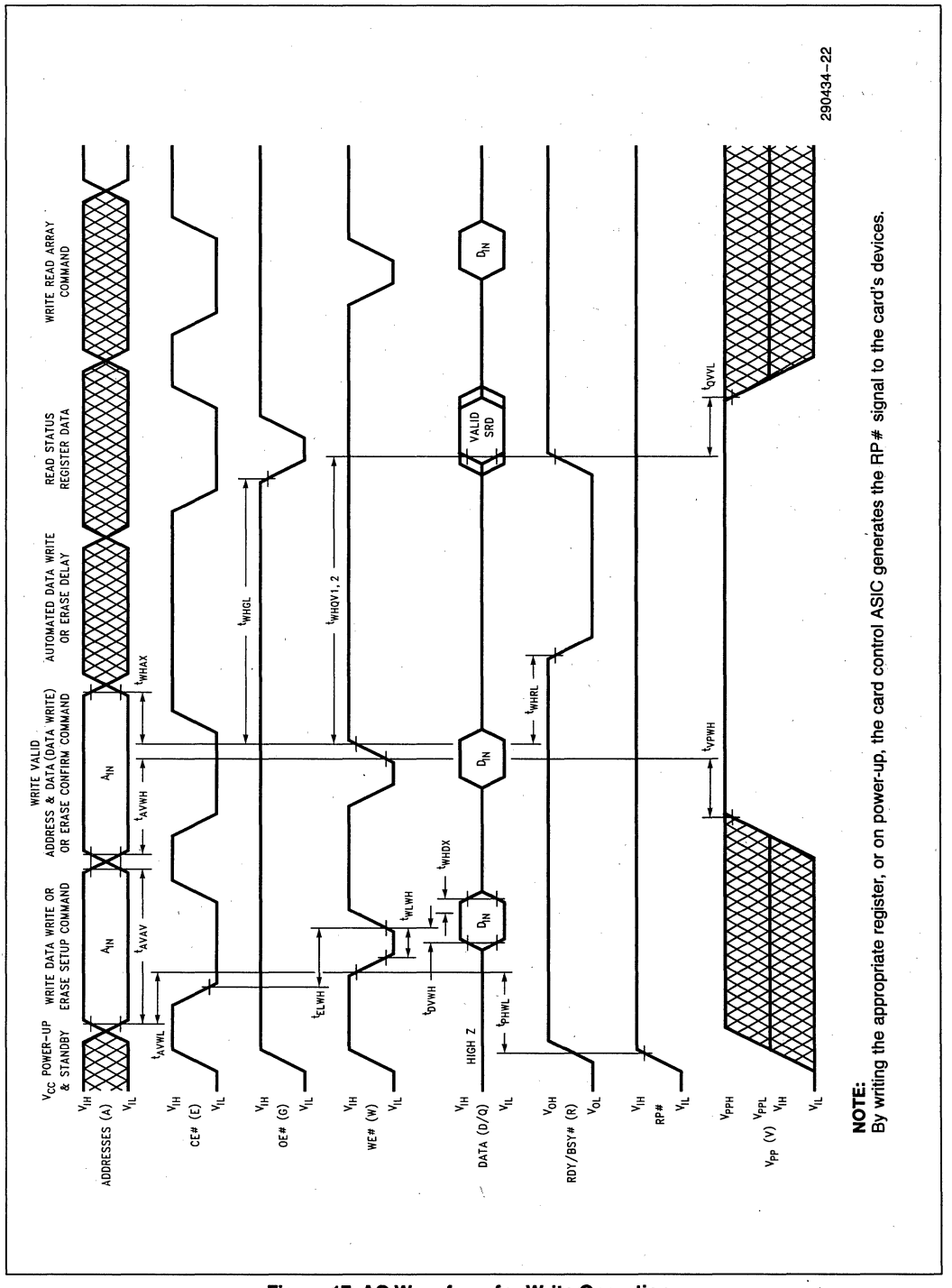
Symbol		Parameter	Notes	Min	Max	Unit
JEDEC	PCMCIA					
t_{AVAV}	t_{WC}	Write Cycle Time		200		ns
t_{WLWH}	t_w (WE)	Write Pulse Width		120		ns
t_{AVWL}	t_{su} (A)	Address Setup Time		20		ns
t_{AVWH}	t_{su} (A-WEH)	Address Setup Time for WE#		140		ns
t_{VPWH}	t_{vps}	V_{PP} Setup to WE# Going High		100		ns
t_{ELWH}	t_{su} (CE-WEH)	Card Enable Setup Time for WE#		140		ns
t_{DVWH}	t_{su} (D-WEH)	Data Setup Time for WE#		60		ns
t_{WHDX}	t_h (D)	Data Hold Time		30		ns
t_{WHAX}	t_{rec} (WE)	Write Recover Time		30		ns
t_{WHRL}		WE# High to RDY/BSY#			120	ns
t_{WHQV1}		Duration of Data Write Operation		6		μs
t_{WHQV2}		Duration of Block Erase Operation		0.3		sec
t_{QVVL}		V_{PP} Hold from Operation Complete	2			ns
t_{WHGL}	t_h (OE-WE)	Write Recovery before Read		10		ns
t_{RHWL}		Reset-PwrDwn Recovery to WE# Going Low		1		μs

NOTES:
1. Read timing characteristics during erase and data write operations are the same as during read-only operations. Refer to AC Characteristics for Read-Only operations.
2. Refer to text on Data-Write and Block-Erase Operations.

BLOCK ERASE AND DATA WRITE PERFORMANCE

Parameter	Notes	Min	Typ[3]	Max	Unit
Block Pair Erase Time[1]	2, 4		1.6	10	sec
Block Pair Write Time	2, 4		0.6	2.1	sec
Byte/Word Write Time	4	6 μs	10 μs	3 ms	

NOTES:
1. Individual blocks can be erased 100,000 times.
2. Excludes System-Level Overhead.
3. 25°C, 12.0 V_{PP}.
4. Monitor Ready/Busy Registers for the completion of a write/erase command.



NOTE:
By writing the appropriate register, or on power-up, the card control ASIC generates the RP# signal to the card's devices.

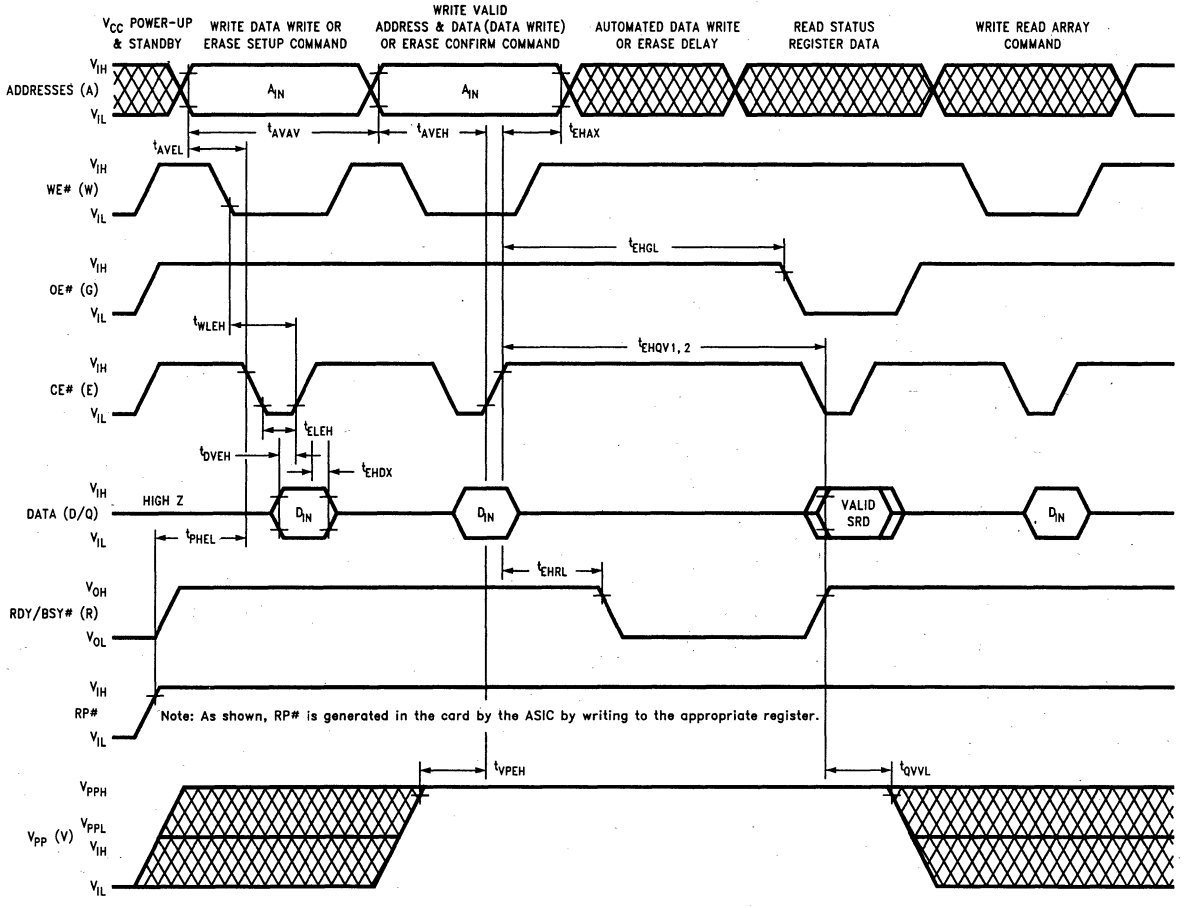
Figure 17. AC Waveform for Write Operations

COMMON AND ATTRIBUTE MEMORY, AC CHARACTERISTICS: CE#-Controlled Write Operations[1]

Symbol		Parameter	Notes	Min	Max	Unit
JEDEC	PCMCIA					
t_{AVAV}	t_{WC}	Write Cycle Time	1	200		ns
t_{ELEH}	t_w (WE)	Chip Enable Pulse Width	1	120		ns
t_{AVEL}	t_{su} (A)	Address Setup Time	1	20		ns
t_{AVEH}	t_{su} (A-WEH)	Address Setup Time for CE#	1	140		ns
t_{VPEH}	t_{vps}	V_{PP} Setup to CE# Going High	1	100		ns
t_{WLEH}	t_{su} (CE-WEH)	Write Enable Setup Time for CE#	1	140		ns
t_{DVEH}	t_{su} (D-WEH)	Data Setup Time for CE#	1	60		ns
t_{EHDX}	t_h (D)	Data Hold Time	1	30		ns
t_{EHAX}	t_{rec} (WE)	Write Recover Time	1	30		ns
t_{EHRL}		CE# High to RDY/BSY#	1		120	ns
t_{EHQV1}	Duration of Data Write	Duration of Data Write Operation	1	6		µs
t_{EHQV2}	Duration of Erase	Duration of Block Erase Operation	1	0.3		sec
t_{QVVL}		V_{PP} Hold from Operation Complete	1, 2	0		ns
t_{EHGL}	t_h (OE-WE)	Write Recovery before Read	1	10		ns
t_{RHEL}		Reset-PwrDwn Recovery to CE# Going Low		1		µs

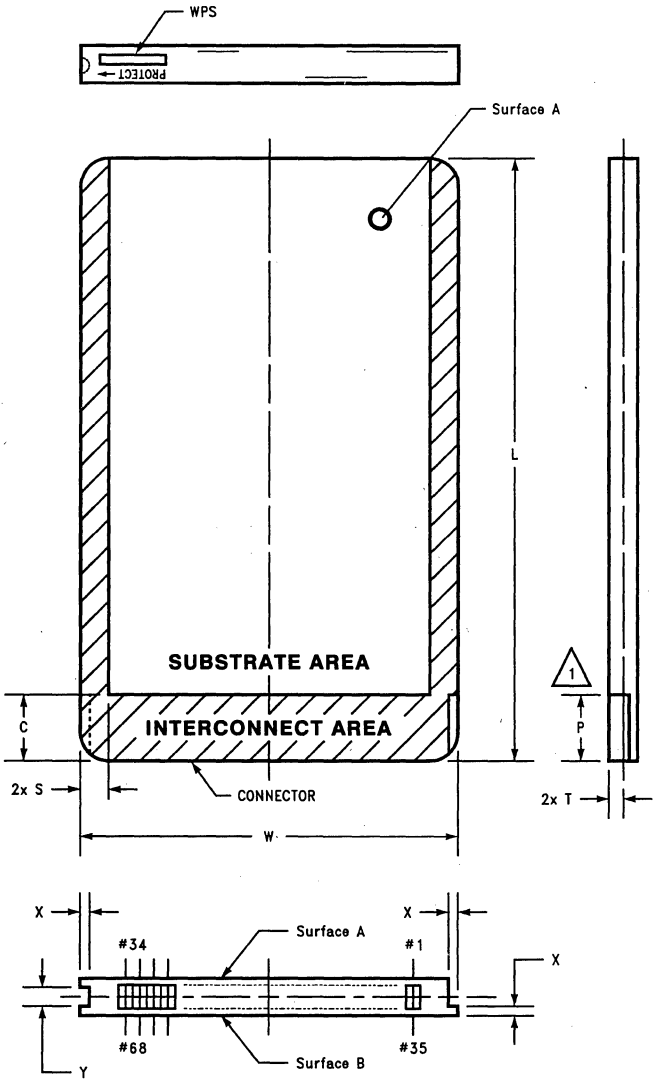
NOTES:
1. Read timing characteristics during erase and data write operations are the same as during read-only operations. Refer to AC Characteristics for Read-Only operations.
2. Refer to text on Data-Write and Block-Erase Operations.

6



NOTE:
By writing the appropriate register, or on power-up, the card control ASIC generates the RP# signal to the card's devices.

Figure 18. Alternate AC Waveform for Write Operations



C MIN	L ± 0.008	P MIN 1	S MIN	T 2	W ± 0.004	X ± 0.002	Y ± 0.002
0.294 (10.0)	3.370 (85.60)	0.394 (10.0)	0.118 (3.0)	0.065 (1.65)	2.126 (54.0)	0.039 (1.00)	0.063 (1.60)

1 POLARIZATION KEY LENGTH.

2 INTERCONNECT AREA TOLERANCE = ±0.002
SUBSTRATE AREA TOLERANCE = ±0.004

3 MILLIMETERS ARE IN PARENTHESIS ().

290434-24

Figure 19. Series 2 Flash Memory Card Package Dimensions

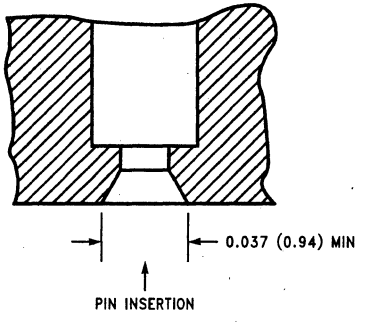
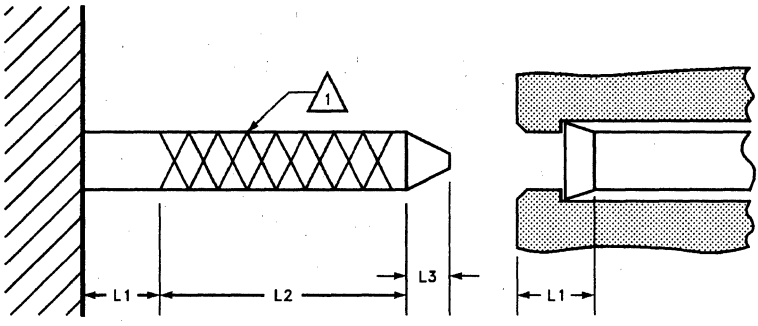


Figure 20. Card Connector Socket



L1 MAX	L2	L3 REF
0.020 (0.5)	Pin Type—See Table 1 Detect 0.059 (1.5) ±0.039 General 0.084 (2.1) ±0.064 Power 0.098 (2.5) ±0.078	0.024 (0.6)

Figure 21. Pin/Socket Contact Length with Wipe

Table 5. Capacitance $T_A = 25°C$, f = 1.0 MHz

Symbol	Characteristics	Commercial		Unit
		Min	Max	
C_{IN}	Address/Control Capacitance (A_0–A_8, CE_1#, CE_2#)		30	pF
	Address/Control Capacitance (A_9–A_{24}, all others)		20	pF
	V_{CC}, V_{PP}		2	μF
C_{OUT}	Output Capacitance		20	pF

NOTE:
Sampled, not 100% tested.

ORDERING INFORMATION

iMC020FLSA,SBXXXXX

WHERE:

i = INTEL
MC = MEMORY CARD
020 = DENSITY IN MEGABYTES
(002,004,010,020 AVAILABLE)
FL = FLASH TECHNOLOGY
S = BLOCKED ARCHITECTURE
A = REVISION
-ET = EXTENDED TEMPERATURE
SBXXXXX = CUSTOMER IDENTIFIER

ADDITIONAL INFORMATION	ORDER NUMBER
28F008SA FlashFile™ Memory Data Sheet	290429
iMC001FLKA 1-Mbyte Flash Memory Card	290399
iMC002FLKA 2-Mbyte Flash Memory Card	290412
iMC004FLKA 4-Mbyte Flash Memory Card	290388
AP-361 "Implementing the Integrated Registers of the Series 2 Flash Memory Card"	292096
AP-364 "28F008SA Automation and Algorithms"	292099
ER-27 "The Intel 28F008SA Flash Memory"	294011
ER-28 "ETOX III Flash Memory Technology"	294012
AP-359 "28F008SA Hardware Interfacing"	292094
AP-360 "28F008SA Software Drivers"	292095

REVISION HISTORY

Number	Description
02	Added 150 ns TUPLE, Deleted 250 ns TUPLE Corrected Global Power Register Address to 4002H Corrected Write Protection Register Address to 4104H Corrected Ready-Busy Mode Register Address to 4140H I_{CC} Standby Byte Wide Mode MAX/TYP Increased Added Power-On Timing Spec Added First Access after Reset Spec Changed Advanced Information to Preliminary
03	Added 2 MByte card support Changed write timing waveforms to match PCMCIA Changed PowerDown (PWD) to Reset-PowerDown (RP)
04	Extended Operating Temperature Range Added Maximum Byte Write Time

iMC004FLKA
4-MBYTE FLASH MEMORY CARD

- **Inherent Nonvolatility (Zero Retention Power)**
  - **— No Batteries Required for Back-up**
- **High-Performance Read**
  - **— 200 ns Maximum Access Time**
- **CMOS Low Power Consumption**
  - **— 40 mA Typical Active Current (X8)**
  - **— 800 μA Typical Standby Current**
- **Flash Electrical Zone-Erase**
  - **— 2 Seconds Typical per 256 Kbyte Zone**
  - **— Multiple Zone-Erase**
- **Random Writes to Erased Zones**
  - **— 10 μs Typical Byte Write**
- **Write Protect Switch to Prevent Accidental Data Loss**
- **Command Register Architecture for Microprocesssor/Microcontroller Compatible Write Interface**
- **ETOX™ II Flash Memory Technology**
  - **— 5V Read, 12V Erase/Write**
  - **— High-Volume Manufacturing Experience**
- **PCMCIA/JEIDA 68-Pin Standard**
  - **— Byte- or Word-wide Selectable**
- **Independent Software & Hardware Vendor Support**
  - **— Integrated System Solution Using Flash Filing Systems**

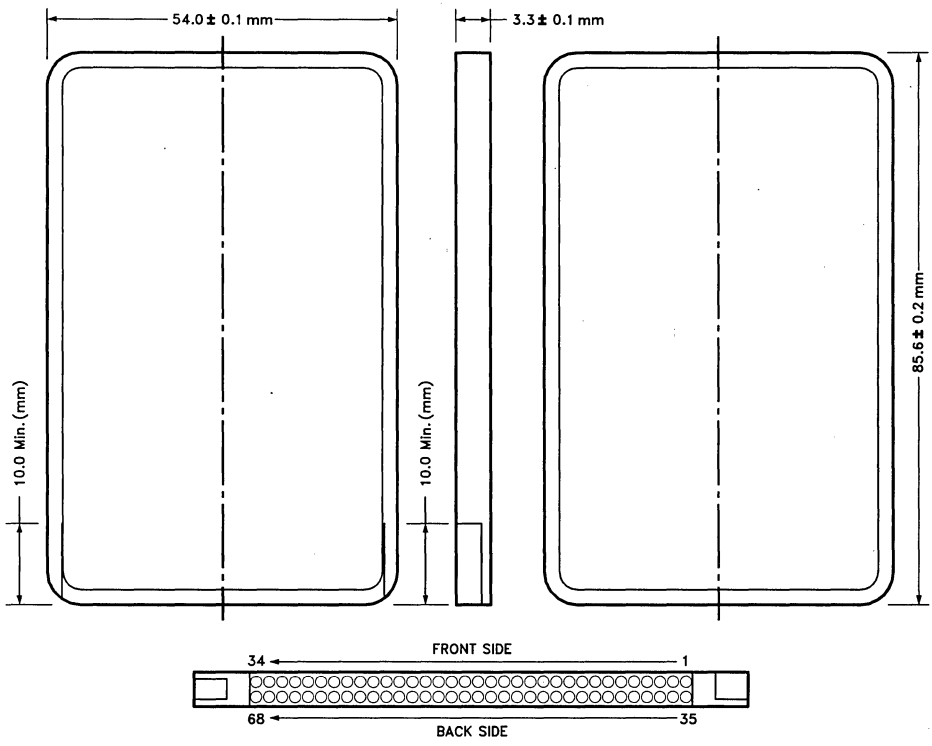
Intel's iMC004FLKA Flash Memory Card is the removable solution for storing and transporting important user data and application code. The combination of rewritability and nonvolatility make the Intel Flash Memory Card ideal for data acquisition and updatable firmware applications. Designing with Intel's Flash Memory Card enables OEM system manufacturers to produce portable and dedicated function systems that are higher performance, more rugged, and consume less power.

The iMC004FLKA conforms to the PCMCIA1.0 international standard, providing standardization at the hardware and data interchange level. OEMs may opt to write the Card Information Structure (CIS) at the memory card's address 00000H with a format utility. This information provides data interchange functional capability. The 200 ns access time allows for "execute-in-place" capability, for popular low-power microprocessors. Intel's 4-MByte Flash Memory Card operates in a byte-wide and word-wide configuration providing performance/power options for different systems.

Intel's Flash Memory card employs Intel's ETOX II Flash Memories. Filing systems, such as Microsoft's* Flash File System (FFS), facilitate data file storage and card erasure using a purely nonvolatile medium in the DOS environment. Flash filing systems, coupled with the Intel Flash Memory Card, effectively create an all-silicon nonvolatile read/write random access memory system that is more reliable and higher performance than disk-based memory systems.

*Microsoft is a trademark of Microsoft Corp.

April 1994
Order Number: 290388-004



290388-1

1	GND	18	V_{PP1}	35	GND	52	V_{PP2}
2	D_3	19	A_{16}	36	CD_1#	53	NC
3	D_4	20	A_{15}	37	D_{11}	54	NC
4	D_5	21	A_{12}	38	D_{12}	55	NC
5	D_6	22	A_7	39	D_{13}	56	NC
6	D_7	23	A_6	40	D_{14}	57	NC
7	CE_1#	24	A_5	41	D_{15}	58	NC
8	A_{10}	25	A_4	42	CE_2#	59	NC
9	OE#	26	A_3	43	NC	60	NC
10	A_{11}	27	A_2	44	NC	61	REG#[1]
11	A_9	28	A_1	45	NC	62	BVD_2#[2]
12	A_8	29	A_0	46	A_{17}	63	BVD_1#[2]
13	A_{13}	30	D_0	47	A_{18}	64	D_8
14	A_{14}	31	D_1	48	A_{19}	65	D_9
15	WE#	32	D_2	49	A_{20}	66	D_{10}
16	NC	33	WP	50	A_{21}	67	CD_2#
17	V_{CC}	34	GND	51	V_{CC}	68	GND

NOTES:
1. REG# = register memory select = No Connect (NC), unused. When REG# is brought low, PCMCIA/JEIDA standard card information structure data is expected. This is accomplished by formatting the card with this data.
2. BVD# = battery detect voltage = Pulled high through pull up resistor.

Figure 1. iMC004FLKA Pin Configurations

Table 1. Pin Description

Symbol	Type	Name and Function
A_0–A_{21}	I	**ADDRESS INPUTS** for memory locations. Addresses are internally latched during a write cycle.
D_0–D_{15}	I/O	**DATA INPUT/OUTPUT:** Inputs data during memory write cycles; outputs data during memory read cycles. The data pins are active high and float to tri-state OFF when the card is deselected or the outputs are disabled. Data is internally latched during a write cycle.
CE_1#, CE_2#	I	**CARD ENABLE:** Activates the card's high and low byte control logic, input buffers, zone decoders, and associated memory devices. CE# is active low; CE# high deselects the memory card and reduces power consumption to standby levels.
OE#	I	**OUTPUT ENABLE:** Gates the cards output through the data buffers during a read cycle. OE# is active low.
WE#	I	**WRITE ENABLE** controls writes to the control register and the array. Write enable is active low. Addresses are latched on the falling edge and data is latched on the rising edge of the WE# pulse. **NOTE:** With $V_{PP} \leq 6.5V$, memory contents cannot be altered.
V_{PP1}, V_{PP2}		**ERASE/WRITE POWER SUPPLY** for writing the command register, erasing the entire array, or writing bytes in the array.
V_{CC}		**DEVICE POWER SUPPLY** (5V ±5%).
GND		**GROUND**
CD_1#, CD_2#	O	**CARD DETECT.** The card is detected when CD_1# and CD_2# = ground.
WP	O	**WRITE PROTECT.** All write operations are disabled with WP = active high.
NC		**NO INTERNAL CONNECTION** to device. Pin may be driven or left floating.
BVD_1#, BVD_2#	O	**BATTERY VOLTAGE DETECT. NOT REQUIRED.**

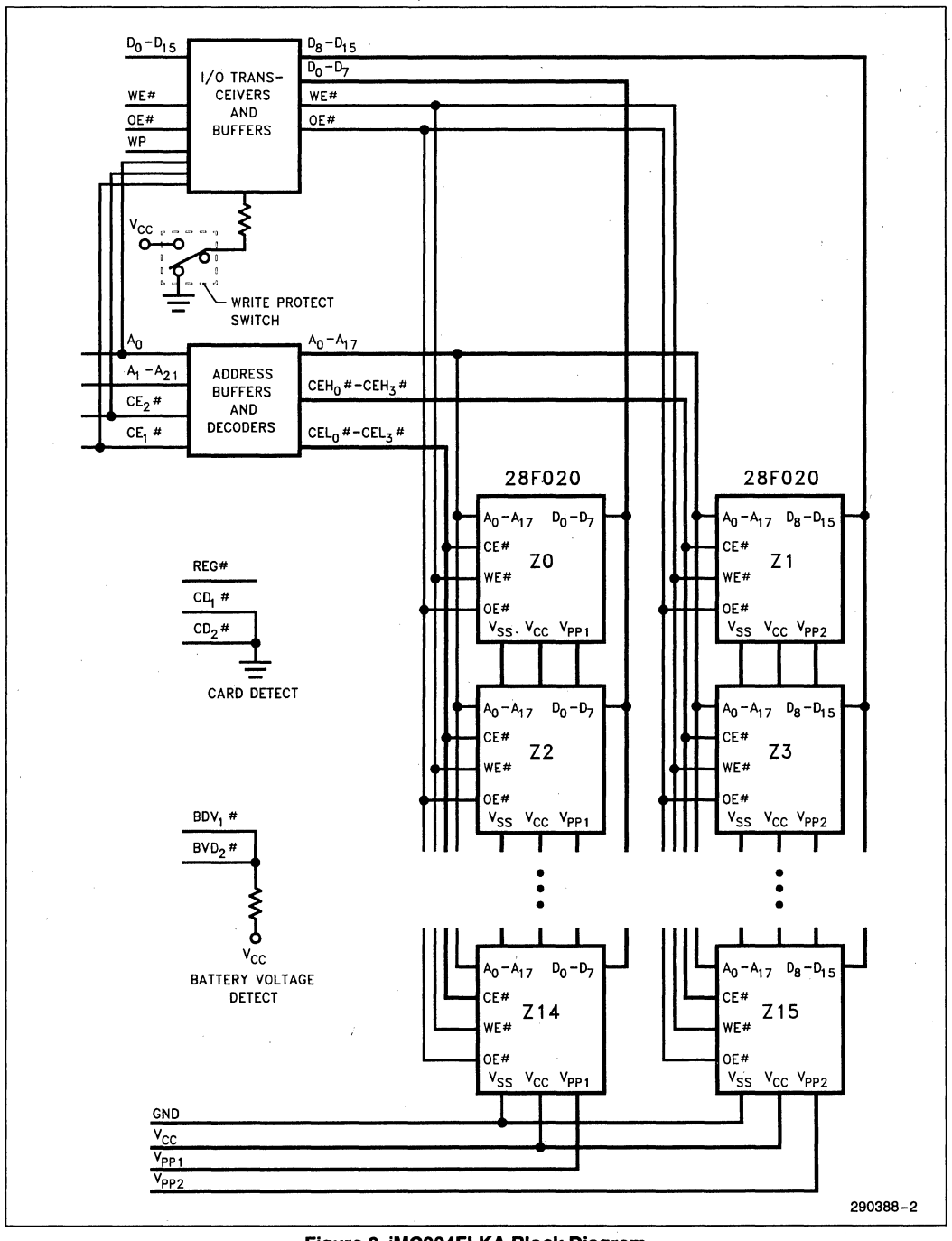


Figure 2. iMC004FLKA Block Diagram

APPLICATIONS

The iMC004FLKA Flash Memory Card allows for the storage of data files and application programs on a purely solid-state removable medium. System resident flash filing systems, such as Microsoft's Flash File System, allow Intel's ETOX II highly reliable Flash Memory Card to effectively function as a physical disk drive.

User application software stored on the flash memory card substantially reduces the slow disk-DRAM download process. Replacing the disk results in a dramatic enhancement of read performance and substantial reduction of active power consumption, size, and weight—considerations particularly important in portable PCs and equipment. The iMC004FLKA's high performance read access time and command register microprocessor write interface allows for use of the flash memory system in an "execute-in-place" architecture. This configuration eliminates the need for the redundancy associated with DRAM and Disk memory system architectures. ROM based operating systems, such as Microsoft's MS-DOS ROM Version allow for "instant-on" capability. This enables the design of PCs that boot, operate, store data files, and execute application code from/to purely nonvolatile memory.

The PCMCIA/JEIDA 68-pin interface with flash filing systems enables the end-user to transport user files and application code between portable PCs and desktop PCs with memory card Reader/Writers. Intel Flash Memory cards provide durable nonvolatile memory storage for Notebook PCs on the road, facilitating simple transfer back into the desktop environment.

For systems currently using a static RAM/battery configuration for data acquisition, the iMC004FLKA's inherent nonvolatility eliminates the need for battery backup. The concern of battery failure no longer exists, an important consideration for portable computers and medical instruments, both requiring continuous operation. The iMC004FLKA consumes no power when the system is off. In addition, the iMC004FLKA offers a considerable cost and density advantage over memory cards based on static RAM with battery backup.

The flash memory card's electrical zone-erasure, byte writability, and complete nonvolatility fit well with data accumulation and recording needs. Electrical zone-erasure gives the designer the flexibility to selectively rewrite zones of data while saving other zones for infrequently updated look-up tables.

PRINCIPLES OF OPERATION

Intel's Flash Memory Card combines the functionality of two mainstream memory technologies: the rewritability of RAM and the nonvolatility of EPROM. The flash memory card consists of an array of individual memory devices, each of which defines a physical zone. The iMC004FLKA's memory devices erase as individual blocks, equivalent in size to the 256 Kbyte zone. Multiple zones can be erased simultaneously provided sufficient current for the appropriate number of zones (memory devices). Note, multiple zone erasure requires higher current from both the V_{PP} and V_{CC} power supplies. Erased zones can then be written in bit- or byte-at-a-time fashion and read randomly like RAM. Bit level write capability also supports disk emulation.

In the absence of high voltage on the $V_{PP1/2}$ pins, the iMC004FLKA remains in the read-only mode. Manipulation of the external memory card-control pin yields the standard read, standby, and output disable operations.

The same read, standby, and output disable operations are available when high voltage is applied to the $V_{PP1/2}$ pins. In addition, high voltage on $V_{PP1/2}$ enables erasure and rewriting of the accessed zone(s). All functions associated with altering zone contents—erase, erase verify, write, and write verify—are accessed via the command register.

Commands are written to the internal memory register(s), decoded by zone size, using standard microprocessor write timings. Register contents for a given zone serve as input to that zone's internal state-machine which controls the erase and rewrite circuitry. Write cycles also internally latch addresses and data needed for write and erase operations. With the appropriate command written to the register(s), standard microprocessor read timings output zone data, or output data for erase and write verification.

Byte-wide or Word-wide Selection

The flash memory card can be read, erased, and written in a byte-wide or word-wide mode. In the word-wide configuration V_{PP1} and/or CE_1# control the LO-Byte while V_{PP2} and CE_2# control the HI-Byte (A_0 = don't care).

Read, Write, and Verify operations are byte- or word-oriented, thus zone independent. Erase Setup and Begin Erase Commands are zone dependent such that commands written to any address within a 256 Kbyte zone boundary initiate the erase operation in that zone (or two 256 Kbyte zones under word-wide operation).

Conventional x8 operation uses CE_1# active-low, with CE_2# high, to read or write data through the D_0–D_7 only. "Even bytes" are accessed when A_0 is low, corresponding to the low byte of the complete x16 word. When A_0 is high, the "odd byte" is accessed by transposing the high byte of the complete x16 word onto the D_0–D_7 outputs. This odd byte corresponds to data presented on D_8–D_{15} pins in x16 mode.

Note that two zones logically adjacent in x16 mode are multiplexed through D_0–D_7 in x8 mode and are toggled by the A_0 address. Thus, zone specific erase operations must be kept discrete in x8 mode by addressing even bytes only for one-half of the zone pair, then addressing odd bytes only for the other half.

Card Detection

The flash memory card features two card detect pins ($CD_{1/2}$#) that allow the host system to determine if the card is properly loaded. Note that the two pins are located at opposite ends of the card. Each CD# output should be read through a port bit. Should only one of the two bits show the card to be present, then the system should instruct the user to re-insert the card squarely into the socket. Card detection can also tell the system whether or not to redirect drives in the case of system booting. $CD_{1/2}$# is active low, internally tied to ground.

Write Protection

The flash memory card features three types of write protection. The first type features a mechanical Write Protect Switch that disables the circuitry that control Write Enable to the flash devices. When the switch is activated, the WE# internal to the card is forced high, which disables any writes to the Command Register. The second type of write protection is based on the PCMCIA/JEIDA socket. Unique pin length assignments provide protective power supply sequencing during hot insertion and removal. The third type operates via software control through the Command Register when the card resides in its connector. The Command Register of each zone is only active when $V_{PP1/2}$ is at high voltage. Depending upon the application, the system designer may choose to make $V_{PP1/2}$ power supply switchable—available only when writes are desired. When $V_{PP1/2}$ = V_{PPL}, the contents of the register default to the read command, making the iMC004FLKA a read-only memory card. In this mode, the memory contents cannot be altered.

The system designer may choose to leave $V_{PP1/2}$ = V_{PPH}, making the high voltage supply constantly available. In this case, all Command Register functions are inhibited whenever V_{CC} is below the write lockout voltage, V_{LKO}. (See the section on Power Up/Down Protection.) The iMC004FLKA is designed to accommodate either design practice, and to encourage optimization of the processor-memory card interface.

BUS OPERATIONS

Read

The iMC004FLKA has two control functions, both of which must be logically active, to obtain data at the outputs. Card Enable (CE#) is the power control and should be used for high and/or low zone(s) selection. Output Enable (OE#) is the output control and should be used to gate data from the output pins, independent of accessed zone selection. In the byte-wide configuration, only one CE# is required. The word-wide configuration requires both CE#s active low.

When $V_{PP1/2}$ is high (V_{PPH}), the read operations can be used to access zone data and to access data for write/erase verification. When $V_{PP1/2}$ is low (V_{PPL}), only read accesses to the zone data are allowed.

Output Disable

With Output Enable at a logic-high level (V_{IH}), output from the card is disabled. Output pins are placed in a high-impedance state.

Standby

With one Card Enable at a logic-high level, the standby operation disables one-half of the x16 output's read/write buffer. Further, only the zone corresponding to the selected address within the upper or lower $CE_{1/2}$# bank is active at a time. (NOTE: A_0 must be low to select the low half of the x16 word when CE_2# = 1 and CE_1# = 0.) All other zones are deselected, substantially reducing card power consumption. For deselected banks, the outputs are placed in a high-impedance state, independent of the Output Enable signal. If the iMC004FLKA is deselected during erasure, writing, or write/erase verification, the accessed zone draws active current until the operation is terminated.

Intelligent Identifier Command

The manufacturer- and device-codes can be read via the Command Register, for instances where the iMC004FLKA is erased and rewritten in a universal reader/writer. Following a write of 90H to a zone's Command Register, a read from address location 00000H on any zone outputs the manufacturer code (89H). A read from address 0002H outputs the memory device code (BDH).

Write

Zone erasure and rewriting are accomplished via the Command Register, when high voltage is applied to $V_{PP1/2}$. The contents of the register serve as input to that zone's internal state-machine. The state-machine outputs dictate the function of the targeted zone.

The Command Register itself does not occupy an addressable memory location. The register is a latch used to store the command, along with address and data information needed to execute the command.

The Command Register is written by bringing Write Enable to a logic-low level (V_{IL}), while Card Enable(s) is/are low. Addresses are latched on the falling edge of Write Enable, while data is latched on the rising edge of the Write Enable pulse. Standard microprocessor write timings are used.

Refer to AC Write Charcteristics and the Erase/Write Waveforms for specific timing parameters.

COMMAND DEFINITIONS

When low voltage is applied to the V_{PP} pin(s), the contents of the zone Command Register(s) default to 00H, enabling read-only operations.

Placing high voltage on the V_{PP} pin(s) enable(s) read/write operations. Zone operations are selected by writing specific data patterns into the Command Register. Tables 3 and 4 define these iMC004FLKA register commands for both byte-wide and word-wide configurations.

All commands written to the Command Register require that the Zone Address be valid or the incorrect zone will receive the command. Any Command/Data Write or Data Read requires the correct Valid Address.

Table 2. Bus Operations

Pins / Operation		Notes	[1, 7] V_{PP2}	[1, 7] V_{PP1}	A0	$CE\#_2$	$CE\#_1$	OE#	WE#	D_8–D_{15}	D_0–D_7
READ-ONLY	Read (x8)	8	V_{PPL}	V_{PPL}	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Even
	Read (x8)	9	V_{PPL}	V_{PPL}	V_{IH}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Odd
	Read (x8)	10	V_{PPL}	V_{PPL}	X	V_{IL}	V_{IH}	V_{IL}	V_{IH}	Data Out	Tri-state
	Read (x16)	11	V_{PPL}	V_{PPL}	X	V_{IL}	V_{IL}	V_{IL}	V_{IH}	Data Out	Data Out
	Output Disable		V_{PPL}	V_{PPL}	X	X	X	V_{IH}	V_{IH}	Tri-state	Tri-state
	Standby		V_{PPL}	V_{PPL}	X	V_{IH}	V_{IH}	X	X	Tri-state	Tri-state
READ/WRITE	Read (x8)	3, 8	V_{PPX}	V_{PPH}	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Even
	Read (x8)	3, 9	V_{PPH}	V_{PPX}	V_{IH}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Odd
	Read (x8)	10	V_{PPH}	V_{PPX}	X	V_{IL}	V_{IH}	V_{IL}	V_{IH}	Data Out	Tri-state
	Read (x16)	3, 11	V_{PPH}	V_{PPH}	X	V_{IL}	V_{IL}	V_{IL}	V_{IH}	Data Out	Data Out
	Write (x8)	5, 8	V_{PPX}	V_{PPH}	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{IL}	Tri-state	Data In-Even
	Write (x8)	9	V_{PPH}	V_{PPX}	V_{IH}	V_{IH}	V_{IL}	V_{IH}	V_{IL}	Tri-state	Data In-Odd
	Write (x8)	10	V_{PPH}	V_{PPX}	X	V_{IL}	V_{IH}	V_{IH}	V_{IL}	Data In	Tri-state
	Write (x16)	11	V_{PPH}	V_{PPH}	X	V_{IL}	V_{IL}	V_{IH}	V_{IL}	Data In	Data In
	Standby	4	V_{PPH}	V_{PPH}	X	V_{IH}	V_{IH}	X	X	Tri-state	Tri-state
	Output Disable		V_{PPH}	V_{PPH}	X	X	X	V_{IH}	V_{IL}	Tri-state	Tri-state

NOTES:
1. Refer to DC Characteristics. When $V_{PP1/2} = V_{PPL}$ memory contents can be read but not written or erased.
2. Manufacturer and device codes may be accessed via a command register write sequence. Refer to Table 3. All other addresses low.
3. Read operations with $V_{PP1/2} = V_{PPH}$ may access array data or the Intelligent Identifier codes.
4. With $V_{PP1/2}$ at high voltage, the standby current equals $I_{CC} + I_{PP}$ (standby).
5. Refer to Table 3 for valid Data-In during a write operation.
6. X can be V_{IL} or V_{IH}.
7. $V_{PPX} = V_{PPH}$ or V_{PPL}.
8. This x8 operation reads or writes the low byte of the x16 word on DQ_{0-7}, i.e., A_0 low reads "even" byte in x8 mode.
9. This x8 operation reads or writes the high byte of the x16 word on DQ_{0-7} (transposed from DQ_{8-15}), i.e., A_0 high reads "odd" byte in x8 mode.
10. This x8 operation reads or writes the high byte of the x16 on DQ_{8-15}. A_0 is "don't care."
11. A_0 is "don't care," unused in x16 mode. High and low bytes are presented simultaneously.

Table 3. Command Definitions Byte-Wide Mode

Command	Notes	Bus Cycles Req'd	First Bus Cycle			Second Bus Cycle		
			Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory		1	Write	RA	00H			
Read Intelligent ID Codes	4	3	Write	IA	90HT	Read		
Set-up Erase/Erase.	5	2	Write	ZA	20H	Write	ZA	20H
Erase Verify	5	2	Write	EA	A0H	Read	EA	EVD
Set-up Write/Write	6	2	Write	WA	40H	Write	WA	WD
Write Verify	6	2	Write	WA	C0H	Read	WA	WVD
Reset	7	2	Write	ZA	FFH	Write	ZA	FFH

Table 4. Command Definitions Word-Wide Mode

Command	Notes	Bus Cycles Req'd	First Bus Cycle			Second Bus Cycle		
			Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory		1	Write	RA	0000H			
Read Intelligent ID Codes	4	3	Write	IA	9090H	Read		
Set-up Erase/Erase	5	2	Write	ZA	2020H	Write	ZA	2020H
Erase Verify	5	2	Write	EA	A0A0H	Read	EA	EVD
Set-up Write/Write	6	2	Write	WA	4040H	Write	WA	WD
Write Verify	6	2	Write	WA	C0C0H	Read	WA	WVD
Reset	7	2	Write	ZA	FFFFH	Write	ZA	FFFFH

NOTES:
1. Bus operations are defined in Table 2.
2. IA = Identifier address: 00H for manufacturer code, 01H for device code.
   EA = Address of memory location to be read during erase verify.
   RA = Read Address
   WA = Address of memory location to be written.
   ZA = Address of 256 Kbyte zones involved in erase operation.
   Addresses are latched on the falling edge of the Write Enable pulse.
3. ID = Data read from location IA during device identification. (Mfr = 89H, Device = BDH).
   EVD = Data read from location EA during erase verify.
   WD = Data to be written at location WA. Data is latched on the rising edge of Write Enable.
   WVD = Data read from location WA during write verify. WA is latched on the Write command.
4. Following the Read inteligent ID command, two read operations access manufacturer and device codes.
5. Figure 5 illustrates the Erase Algorithm.
6. Figure 6 illustrates the Write Algorithm.
7. The second bus cycle must be followed by the desired command register write.
8. The Reset command operation on a zone basic, To reset entire Card, requires reset write cycles to each zone.

Read Command

While $V_{PP1/2}$ is high, for erasure and writing, zone memory contents can be accessed via the read command. The read operation is initiated by writing 00H (0000H for the word-wide configuration) into the zone Command Register(s). Microprocessor read cycles retrieve zone data. The accessed zone remains enabled for reads until the Command Register(s) contents are altered.

The default contents of each zone's register(s) upon $V_{PP1/2}$ power-up is 00H (00000H for word-wide). This default value ensures that no spurious alteration of memory card contents occurs during the $V_{PP1/2}$ power transition. Where the $V_{PP1/2}$ supply is left at V_{PPH}, the memory card powers-up and remains enabled for reads until the command Register contents of targeted zones are changed. Refer to the AC Read Characteristics and Waveforms for specific timing parameters.

Intelligent Identifier Command

Each zone of the iMC004FLKA contains an Intelligent Identifier to identify memory card device characteristics. The operation is initiated by writing 90H (9090H for word-wide) into the Command Register(s) with Zone Address. Following the command write, a read cycle from address 00000H retrieves the manufacturer code 89H (8989H for word-wide). A read cycle from address 0002H returns the device code BDH (BDBDH for word-wide). To terminate the operation, it is necessary to write another valid command into the register(s).

Set-up Erase/Erase Commands

Set-up Erase stages the targeted zone(s) for electrical erasure of all bytes in the zone. The set-up erase operation is performed by writing 20H to the Command Register (2020H for word-wide) with Zone Address.

To commence zone-erasure, the erase command (20H or 2020H) must again be written to the register(s). The erase operation begins with the rising edge of the Write-Enable pulse and terminates with the rising edge of the next Write-Enable pulse (i.e., Erase-Verify Command with zone address).

This two-step sequence of set-up followed by execution ensures that zone memory contents are not accidentally erased. Also, zone-erasure can only occur when high voltage is applied to the $V_{PP1/2}$ pins. In the absence of this high voltage, zone memory contents are protected against erasure. Refer to AC Erase Characterstics and Waveforms for specific timing parameters.

Erase-Verify Command

The erase command erases all of the bytes of the zone in parallel. After each erase operation, all bytes in the zone must be individually verified. In byte-mode operations, zones are segregated by A_0 in odd and even banks; erase and erase verify operations must be done in complete passes of even-bytes-only then odd-bytes-only. See the Erase Algorithm for byte-wide mode. The erase verify operation is initiated by writing A0H (A0A0H for word-wide) into the Command Register(s). The address for the byte(s) to be verified must be supplied as it is latched on the falling edge of the Write Enable pulse. The register write terminates the erase operation with the rising edge of its Write Enable pulse.

The enabled zone applies an internally-generated margin voltage to the addressed byte. Reading FFH from the addressed byte indicates that all bits in the byte are erased. Similarly, reading FFFFH from the addressed word indicates that all bits in the word are erased.

The erase-verify command must be written to the Command Register prior to each byte (word) verification to latch its address. The process continues for each byte (word) in the zone(s) until a byte (word) does not return FFH (FFFFH) data, or the last address is accessed.

In the case where the data read is not FFH (FFFFH), another erase operation is performed. (Refer to Set-up Erase/Erase.) Verification then resumes from the address of the last-verified byte (word). Once all bytes (words) in the zone(s) have been verified, the erase step is complete. The accessed zone can now be written. At this point, the verify operation is terminated by writing a valid command (e.g., Write Set-up) to the Command Register. The Erase algorithms for byte-wide and word-wide configurations illustrate how commands and bus operations are combined to perform electrical erasure of the iMC001FLKA. Refer to AC Erase Characteristics and Waveforms for specific timing parameters.

Set-up Write/Write Commands

Set-up write is a command-only operation that stages the targeted zone for byte writing. Writing 40H (4040H) into the Command Register(s) performs the set-up operation.

Once the write set-up operation is performed, the next Write Enable pulse causes a transition to an active write operation. Addresses are internally latched on the falling edge of the Write Enable pulse. Data is internally latched on the rising edge of the Write Enable pulse. The rising edge of Write Enable also begins the write operation. The write operation terminates with the next rising edge of Write Enable, which is used to write the verify command. Refer to AC Write Characteristics and Waveforms for specific timing parameters.

Write Verify Command

The iMC004FLKA is written on a byte-by-byte or word-by-word basis. Byte or word writing may occur sequentially or at random. Following each write operation, the byte or word just written must be verified.

The write-verify operation is initiated by writing C0H (C0C0H) into the Command Register(s) with the correct address. The register write(s) terminate(s) the write operation with the rising edge of its Write Enable pulse. The write-verify operation stages the accessed zone(s) for verification of the byte or word last written. The zone(s) apply(ies) an internally-generated margin voltage to the byte or word. A microprocessor read cycle outputs the data. A successful comparison between the written byte or word and true data means that the byte or word is successfully written. The write operation then proceeds to the next desired byte or word location. The Write algorithms for byte-wide and word-wide configurations illustrate how commands are combined with bus operations to perform byte and word writes. Refer to AC Write Characteristics and Waveforms for specific timing parameters.

Reset Command

A reset command is provided as a means to safely abort the erase- or write-command sequences. Following either set-up command (erase or write) with two consecutive writes of FFH (FFFFH for word-wide) will safely abort the operation. Zone memory contents will not be altered. A valid command must then be written to place the accessed zone in the desired state.

EXTENDED ERASE/WRITE CYCLING

Intel has designed extended cycling capability into its ETOX II flash memory technology enabling a flash memory card with a MTBF that is many times more reliable than rotating disk technology. Resulting improvements in cycling reliability come without increasing memory cell size or complexity. First, an advanced tunnel oxide increases the charge carrying ability ten-fold. Second, the oxide area per cell subjected to the tunneling electric field minimizes the probability of oxide defects in the region. The lower electric field greatly reduces oxide stress and the probability of failure.

WRITE ALGORITHMS

The write algorithm(s) use write operations of 10 μs duration. Each operation is followed by a byte or word verification to determine when the addressed byte or word has been successfully written. The algorithm(s) allows for up to 25 write operations per byte or word, although most bytes and words verify on the first or second operation. The entire sequence of writing and byte/word verification is performed with V_{PP} at high voltage.

ERASE ALGORITHM

The Erase algorithm(s) yield(s) fast and reliable electrical erasure of memory contents. The algorithm employs a closed-loop flow, similar to the write algorithm, to simultaneously remove charge from all bits in the accessed zone(s).

Erasure begins with a read of memory zone contents. Reading FFH (FFFFH) data from the accessed zone(s) can be immediately followed by writing to the desired zone(s).

For zones being erased and rewritten, uniform and reliable erasure is ensured by first writing all bits in the accessed zone to their charged state (data = 00H byte-wide, 00000H word-wide). This is accomplished, using the write algorithm, in approximately four seconds per zone.

Erase execution then continues with an initial erase operation. Erase verification (data = FFH byte-wide, FFFFH word-wide) begins at address 00000H and continues through the zone to the last address, or until data other than FFH (FFFFH) is encountered. (Note: byte-wide erase operation requires separate even- and odd-address passes to handle the individual 256 Kbyte zones.) With each erase operation, an increasing number of bytes or words verify to the erased state. Erase efficiency may be improved by storing the address of the last byte or word verified in a register(s). Following the next erase operation, verification starts at the stored address location. Follow this procedure until all bytes in the zone are erased. Then, re-start the procedure for the next zone or word-wide zone pair. Erasure typically occurs in two seconds per zone.

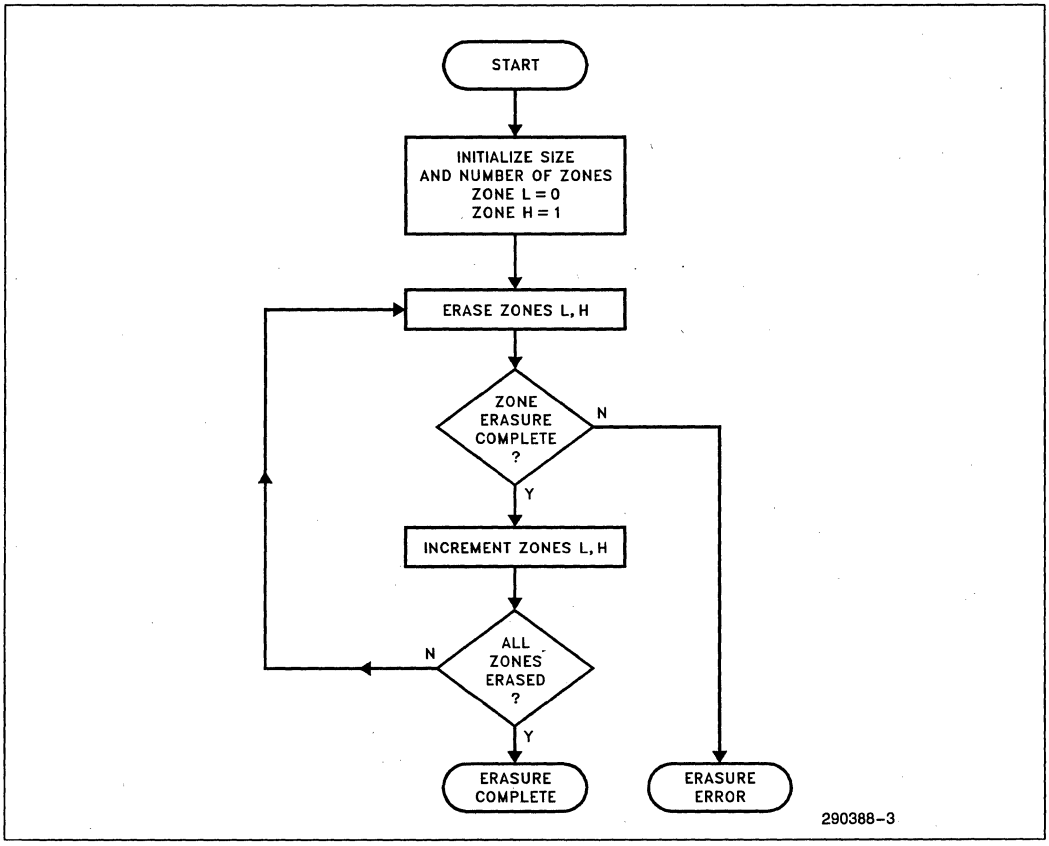
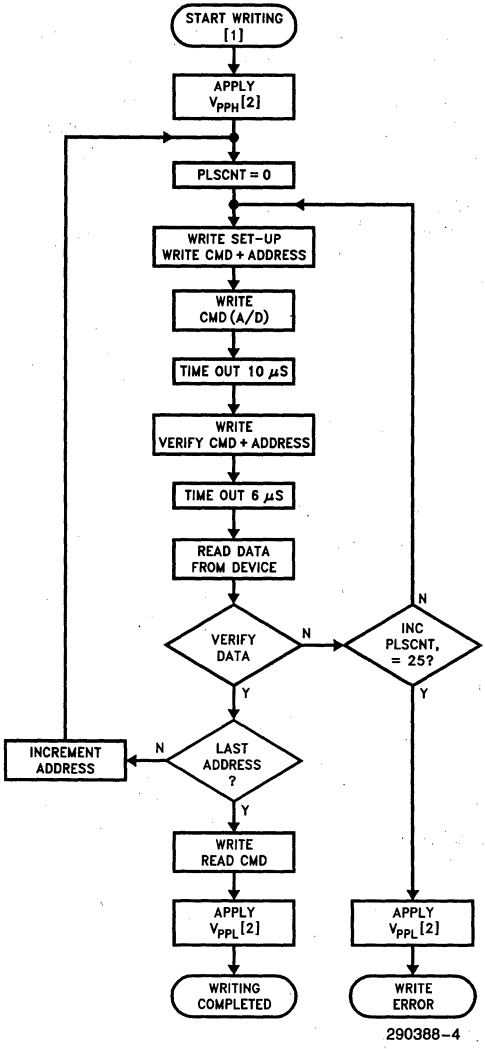


Figure 3. Full Card Erase Flow

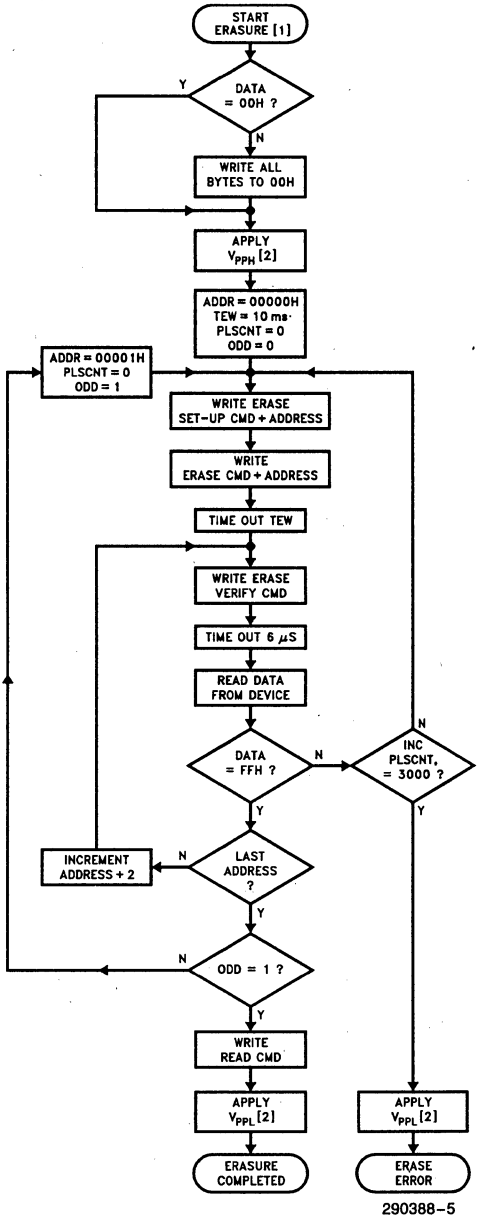
6



Bus Operation	Command	Comments
Standby		Wait for V_{PP} ramp to V_{PPH} (= 12.0V) (2)
		Initialize pulse-count
Write	Set-up Write	Data = 40H + Valid Address
Write	Write	Valid address/data
Standby		Duration of Write Operation (t_{WHWH1})
Write	Write(3) Verify	Data = C0H at Valid Address; Stops (4) Write Operation
Standby		t_{WHGL}
Read		Read byte to verify Write Operation at Valid Address
Standby		Compare data output to data expected
Write	Read	Data = 00H, resets the register for read operations.
Standby		Wait for V_{PP} ramp to V_{PPL} (2)

NOTES:
1. CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.
2. See DC Characteristics for the value of V_{PPH} and V_{PPL}.
3. Write Verify is only performed after a byte write operation. A final read/compare may be performed (optional) after the register is written with the Read command.
4. Refer to principles of operation.

Figure 4. Write Algorithm for Byte-Wide Mode



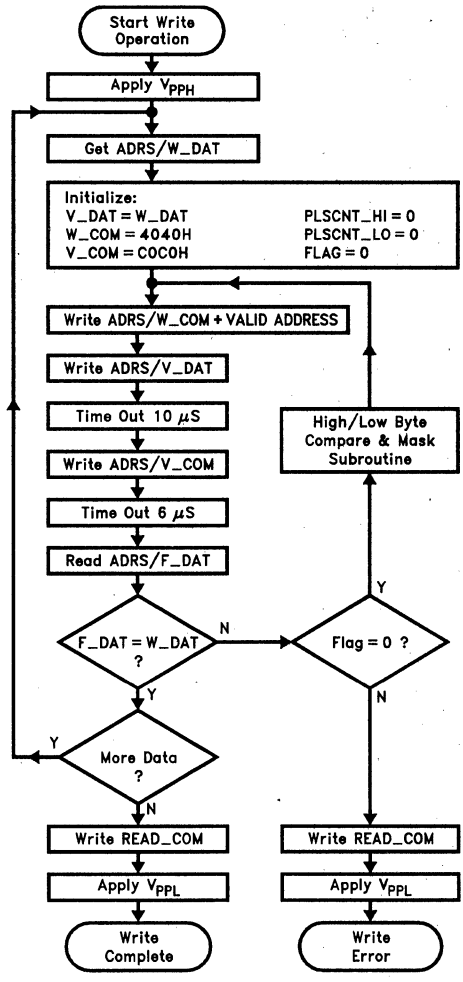
290388-5

Bus Operation	Command	Comments
Standby		Wait for V_{PP} ramp to V_{PPH} (= 12.0V)(2)
		Use with Write Operation Algorithm
		Initialize even/odd Addresses, Erase Pulse Width, and Pulse Count
Write	Set-up Erase	Data = 20H + Address
Write	Erase	Data = 20H + Address
Standby		Duration of Erase operation (t_{WHWH2})
Write	Erase Verify(3)	Addr = Byte to verify; Data = A0H; Stops
Standby		Erase Operation(4) t_{WHGL}
Read		Read byte to verify erasure at address
Standby		Compares output to FFH increment pulse count
Write	Read	Data = 00H, resets the register for read operations.
Standby		Wait for V_{PP} ramp to V_{PPL}(2)

NOTES:
1. CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.
2. See DC Characteristics for the value of V_{PPH} and V_{PPL}.
3. Erase Verify is only performed after a chip erasure. A final read/compare may be performed (optional) after the register is written with the Read command.
4. Refer to principles of operation.

Figure 5. Erase Algorithm for Byte-Wide Mode

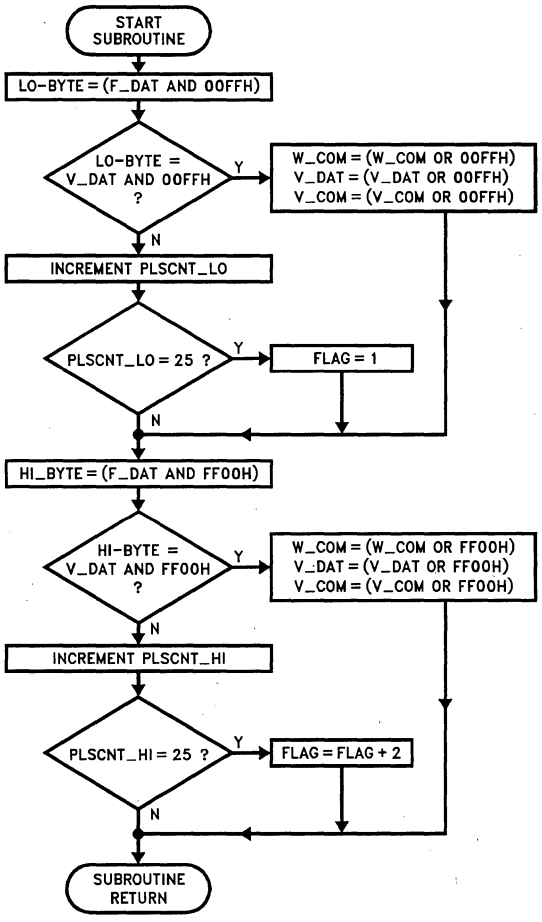
6



Comments
Wait for V_{PP} ramp to V_{PPH}
ADRS = address to write W_DAT = data word to write
Initialize Data Word Variables: V_DAT = valid data W_COM = Write Command V_COM = Write Verify Command PLSCNT_HI = HI Byte Pulse Counter PLSCNT_LO = LO Byte Pulse Counter FLAG = Write Error Flag
Write Set-up Command Address needs to be Valid Write
See Write Verify and Mask Subroutine
Write Verify Command
F_DAT = flash memory data
Compare flash memory data to valid data (word compare). If not equal, check for Write Error flag. If Flag not set, compare High and Low Bytes in the Subroutine.
Check buffer of I/O port for more data to write
Reset device for read operation
Turn off V_{PP}

290388-6

Figure 6. Write Algorithm for Word-Wide Mode



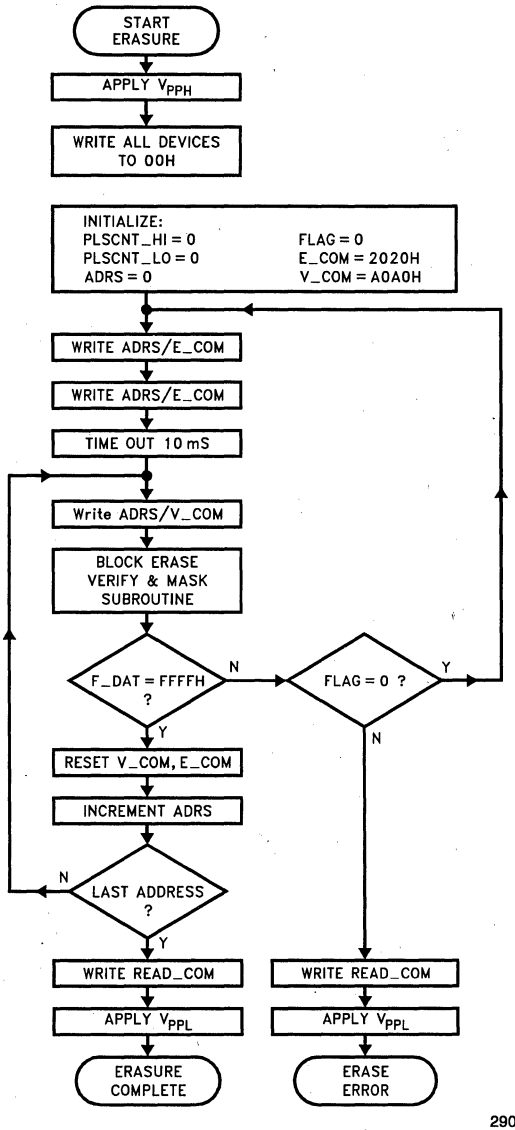
290388-7

Comments
To look at the LO Byte, Mask* the HI Byte with 00
If the LO Byte verifies, mask the LO Byte commands with the reset command (FFH)
If the LO Byte does not verify, then increment its pulse counter and check for max count FLAG = 1 denotes a LO Byte error
Repeat the sequence for the HI Byte
FLAG = 2 denotes a HI Byte error FLAG = 3 denotes both a HI and LO Byte errors. Flag = 0 denotes no max count errors; continue with algorithm.

*Masking can easily and efficiently be done in assembly languages. Simply load word registers with the incoming data (F__DAT), the program commands and the verify commands. Then manipulate the HI or LO register contents.

Figure 7. Write Verify and Mask Subroutine for Word-Wide Mode

6

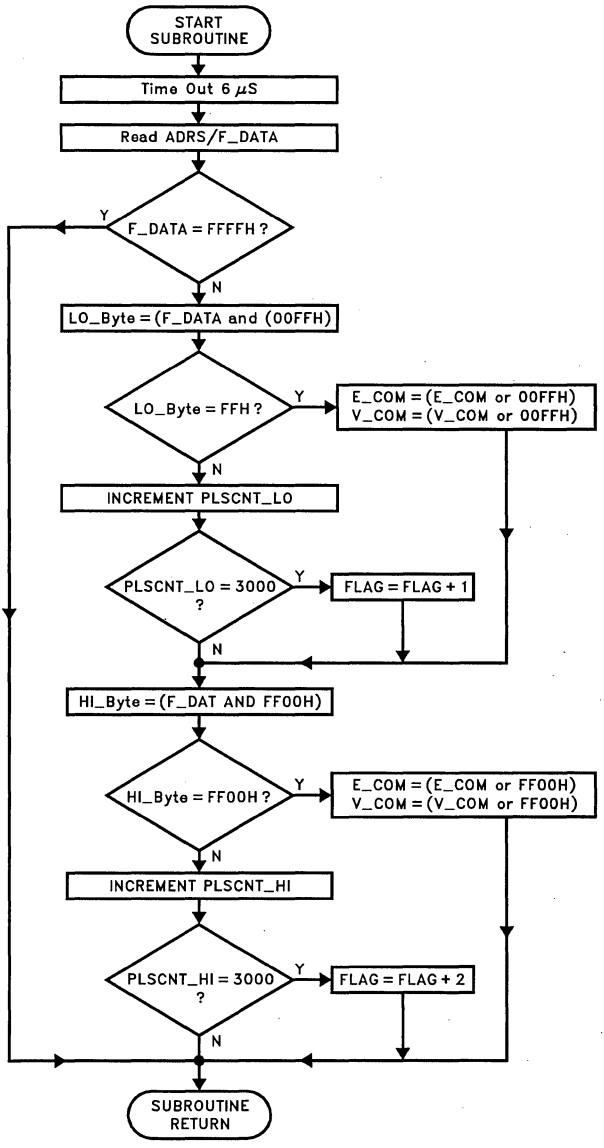


290388-8

Comments
Wait for V_{PP} to stabilize.
Use Write operation algorithm in x8 or x16 configuration
Initialize Variables: PLSCNT_HI = HI Byte Pulse Counter PLSCNT_LO = LO Byte Pulse Counter FLAG = Erasure error flag ADRS = Address E_COM = Erase Command V_COM = Verify Command
Erase Set-up Command
Start Erasing
Duration of Erase Operation
Erase Verify Command stops erasure
See Block Erase Verify & Mask Subroutine
When both devices at ADRS are erased, F_DATA = FFFFH. If not equal, increment the pulse counter and check for last pulse
Reset commands default to (E_COM = 2020H) (V_COM = A0A0H) before verifying next ADRS
Reset device for read operation
Turn off V_{PP}

NOTE:
X16 Addressing uses A_1–A_{21} only. A_0 = 0 throughout word-wide operation.

Figure 8. Erase Algorithm for Word-Wide Mode



290388-9

Comments
This subroutine reads the data word (F__DATA). It then masks the HI or LO Byte of the Erase and Verify commands from executing during the next operation.
If both HI and LO Bytes verify, then return.
Mask* the HI Byte with 00H.
If the LO Byte verifies erasure, then mask* the next erase and verify commands with FFH (RESET).
If the LO Byte does not verify, then increment its pulse counter and check for max count. FLAG = 1 denotes a LO Byte error.
Repeat the sequence for the HI Byte.
Flag = 2 denotes a HI Byte error. Flag = 3 denotes both a HI and LO Byte errors. FLAG = 0 denotes no max count errors; continue with algorithm.

*Masking can easily and efficiently be done in assembly languages. Simply load word registers with the incoming data (F__DAT), the program commands and the verify commands. Then manipulate the HI or LO register contents.

Figure 9. Erase Verify and Mask Subroutine for Word-Wide Mode

6

SYSTEM DESIGN CONSIDERATIONS

Three-Line Control

Three-line control provides for:

a. the lowest possible power dissipation and.

b. complete assurance that output bus contention will not occur.

To efficiently use these three control inputs, an address-decoder output should drive $CE_{1,\ 2}$#, while the system's Read signal controls the card OE# signal, and other parallel zones. This, coupled with the internal zone decoder, assures that only enabled memory zones have active outputs, while deselected zones maintain the low power standby condition.

Power-Supply Decoupling

Flash memory power-switching characteristics require careful device decoupling. System designers are interested in three supply current issues—standby, active and transient current peaks, produced by falling and rising edges of $CE_{1/2}$#. The capacitive and inductive loads on the card and internal flash memory zones determine the magnitudes of these peaks.

Three-line control and proper decoupling capacitor selection will suppress transient voltage peaks. The iMC004FLKA features on-card ceramic decoupling capacitors connected between V_{CC} and V_{SS}, and between V_{PP1}/V_{PP2} and V_{SS}.

The card connector should also have a 4.7 μF electrolytic capacitor between V_{CC} and V_{SS}, as well as between V_{PP1}/V_{PP2} and V_{SS}. The bulk capacitors will overcome voltage slumps caused by printed-circuit-board trace inductance, and will supply charge to the smaller capacitors as needed.

Power Up/Down Protection

The PCMCIA/JEIDA socket is specified, via unique Pin lengths, to properly sequence the power supplies to the flash memory card. This assures that hot insertion and removal will not result in card damage or data loss.

Each zone in the iMC004FLKA is designed to offer protection against accidental erasure or writing, caused by spurious system-level signals that may exist during power transitions. The card will power-up into the read state.

A system designer must guard against active writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and $CE_{1,\ 2}$# must be low for a command write, driving either to V_{IH} will inhibit writes. With its control register architecture, alteration of zone contents only occurs after successful completion of the two-step command sequences.

While these precautions are sufficient for most applications, it is recommended that V_{CC} reach its steady state value before raising $V_{PP1/2}$ above V_{CC} + 2.0V. In addition, upon powering-down, $V_{PP1/2}$ should be below V_{CC} + 2.0V, before lowering V_{CC}.

Absolute Maximum Ratings*

Operating Temperature
During Read ..... 0°C to + 60°C(1)
During Erase/Write ..... 0°C to + 60°C
Temperature Under Bias ..... −10°C to +70°C
Storage Temperature ..... −30°C to +70°C
Voltage on Any Pin with
Respect to Ground ..... −2.0V to +7.0V(2)
V_{PP1}/V_{PP2} Supply Voltage with
Respect to Ground
During Erase/Write ..... −2.0V to +14.0V(2, 3)
V_{CC} Supply Voltage with
Respect to Ground ..... −2.0V to +7.0V(2)

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for commercial product defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to −2.0V for periods less than 20 ns. Maximum DC voltage on output pins is V_{CC} + 0.5V, which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.
3. Maximum DC input voltage on V_{PP1}/V_{PP2} may overshoot to +14.0V for periods less than 20 ns.

OPERATING CONDITIONS

Symbol	Parameter	Limits		Unit	Comments
		Min	Max		
T_A	Operating Temperature	0	60	°C	For Read-Only and Read/Write Operations
V_{CC}	V_{CC} Supply Voltage	4.75	5.25	V	
V_{PPH}	Active V_{PP1}, V_{PP2} Supply Voltages	11.40	12.60	V	
V_{PPL}	V_{PP} During Read Only Operations	0.00	6.50	V	

6

DC CHARACTERISTICS—Byte Wide Mode

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1, 4		±1.0	±20	μA	$V_{CC} = V_{CC}$ max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1		±1.0	±20	μA	$V_{CC} = V_{CC}$ max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1		0.8	1.6	mA	$V_{CC} = V_{CC}$ max, CE = V_{CC} ±0.2V
				4	7	mA	CE# = V_{IH}, $V_{CC} = V_{CC}$ max
I_{CC1}	V_{CC} Active Read Current	1, 2		40	70	mA	$V_{CC} = V_{CC}$ max CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Write Current	1, 2		5.0	15	mA	Writing in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		10	20	mA	Erasure in Progress
I_{CC4}	V_{CC} Write Verify Current	1, 2		10	20	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress

DC CHARACTERISTICS—Byte Wide Mode (Continued)

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{CC5}	V_{CC} Erase Verify Current	1, 2		10	20	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±80	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1, 3		0.7	1.6	mA	$V_{PP} > V_{CC}$
					±0.08		$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Write Current	1, 3		8.0	30	mA	$V_{PP} = V_{PPH}$ Write in Progress
I_{PP3}	V_{PP} Erase Current	1, 3		10	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Write Verify Current	1, 3		3.0	6.0	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 3		3.0	6.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.4		V_{CC} ± 0.3	V	
V_{OL}	Output Low Voltage				0.40	V	$I_{OL} = 3.2$ mA $V_{CC} = V_{CC}$ min
V_{OH1}	Output High Voltage		3.8			V	$I_{OH} = -2.0$ mA $V_{CC} = V_{CC}$ min
V_{PPL}	V_{PP} During Read-Only Operations		0.00		6.5	V	Note: Erase/Write are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} During Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at $V_{CC} = 5.0V$, $V_{PP} = 12.0V$, $T = 25°C$.
2. 1 chip active and 15 in standby for byte-wide mode.
3. Assumes 1 V_{PP} is active.
4. Due to 100 kΩ pull up resistors, OE#, CE_1#, CE_2#, and WE# will exhibit ≤ 55 μA of additional I_{LI} when $V_{IN} = V_{SS}$.

DC CHARACTERISTICS—Word Wide Mode

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1, 4		±1.0	±20	μA	$V_{CC} = V_{CC}$ max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1		±1.0	±20	μA	$V_{CC} = V_{CC}$ max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1		0.8	1.6	mA	$V_{CC} = V_{CC}$ max, CE# = V_{CC} ±0.2V
				4	7	mA	CE# = V_{IH}, $V_{CC} = V_{CC}$ max

DC CHARACTERISTICS—Word Wide Mode (Continued)

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{CC1}	V_{CC} Active Read Current	1, 2		50	100	mA	$V_{CC} = V_{CC}$ max CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Write Current	1, 2		5.0	25	mA	Writing in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		15	30	mA	Erasure in Progress
I_{CC4}	V_{CC} Write Verify Current	1, 2		15	30	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		15	30	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±160	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current	1, 3		1.5	3.0	mA	$V_{PP} > V_{CC}$
	or Standby Current				±.16		$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Write Current	1, 3		17	63	mA	$V_{PP} = V_{PPH}$ Write in Progress
I_{PP3}	V_{PP} Erase Current	1, 3		20	60	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Write Verify Current	1, 3		5.0	12	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 3		5.0	12	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.4		V_{CC} ± 0.3	V	
V_{OL}	Output Low Voltage				0.40	V	I_{OL} = 3.2 mA $V_{CC} = V_{CC}$ min
V_{OH1}	Output High Voltage		3.8			V	I_{OH} = −2.0 mA $V_{CC} = V_{CC}$ min
V_{PPL}	V_{PP} During Read-Only Operations		0.00		6.5	V	Note: Erase/Write are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} During Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. 2 chips active and 14 in standby for word-wide mode.
3. Assumes 2 V_{PPS} are active.
4. Due to 100 kΩ pull up resistors, OE#, CE_1#, CE_2#, and WE# will exhibit ≤ 55 μA of additional I_{LI} when $V_{IN} = V_{SS}$.

CAPACITANCE $T = 25°C, f = 1.0$ MHz

Symbol	Parameter	Notes	Limits		Unit	Conditions
			Min	Max		
C_{IN1}	Address Capacitance			40	pF	$V_{IN} = 0V$
C_{IN2}	Control Capacitance			40	pF	$V_{IN} = 0V$
C_{OUT}	Output Capacitance			40	pF	$V_{OUT} = 0V$
$C_{I/O}$	I/O Capacitance			40	pF	$V_{I/O} = 0V$

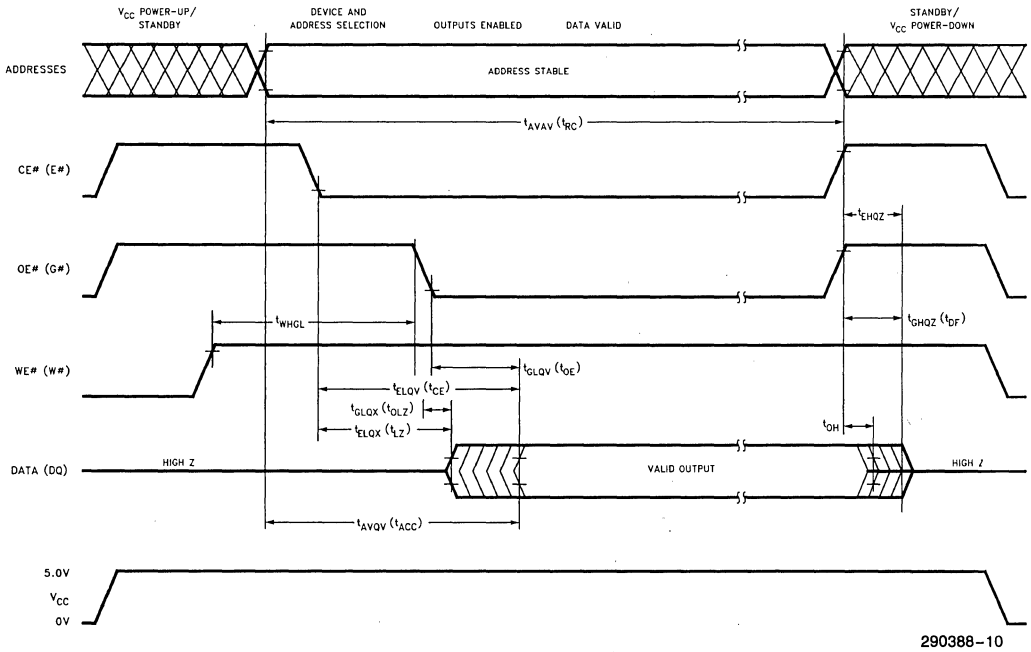
AC TEST CONDITIONS

Input Rise and Fall Times (10% to 90%) . . . . . 10 ns
Input Pulse Levels . . . . . V_{OL} and V_{OH1}
Input Timing Reference Level . . . . . V_{IL} and V_{IH}
Output Timing Reference Level . . . . . V_{IL} and V_{IH}

AC CHARACTERISTICS—Read-Only Operations

Symbol	Characteristic	Notes	Min	Max	Unit
t_{AVAV}/t_{RC}	Read Cycle Time	2	200		ns
t_{ELQV}/t_{CE}	Chip Enable Access Time	2		200	ns
t_{AVQV}/t_{ACC}	Address Access Time	2		200	ns
t_{GLQV}/t_{OE}	Output Enable Access Time	2		100	ns
t_{ELQX}/t_{LZ}	Chip Enable to Output in Low Z	2	5		ns
t_{EHQZ}	Chip Disable to Output in High Z	2		60	ns
t_{GLQX}/t_{OLZ}	Output Enable to Output in Low Z	2	5		ns
t_{GHQZ}/t_{DF}	Output Disable to Output in High Z	2		60	ns
t_{OH}	Output Hold from Address, CE#, or OE# Change	1, 2	5		ns
t_{WHGL}	Write Recovery Time before Read	2	6		μs

NOTES:
1. Whichever occurs first.
2. Rise/Fall Time ≤ 10 ns.



NOTE:
CE# refers to $CE_{1,2}$#.

Figure 10. AC Waveforms for Read Operations

6

AC CHARACTERISTICS—For Write/Erase Operations

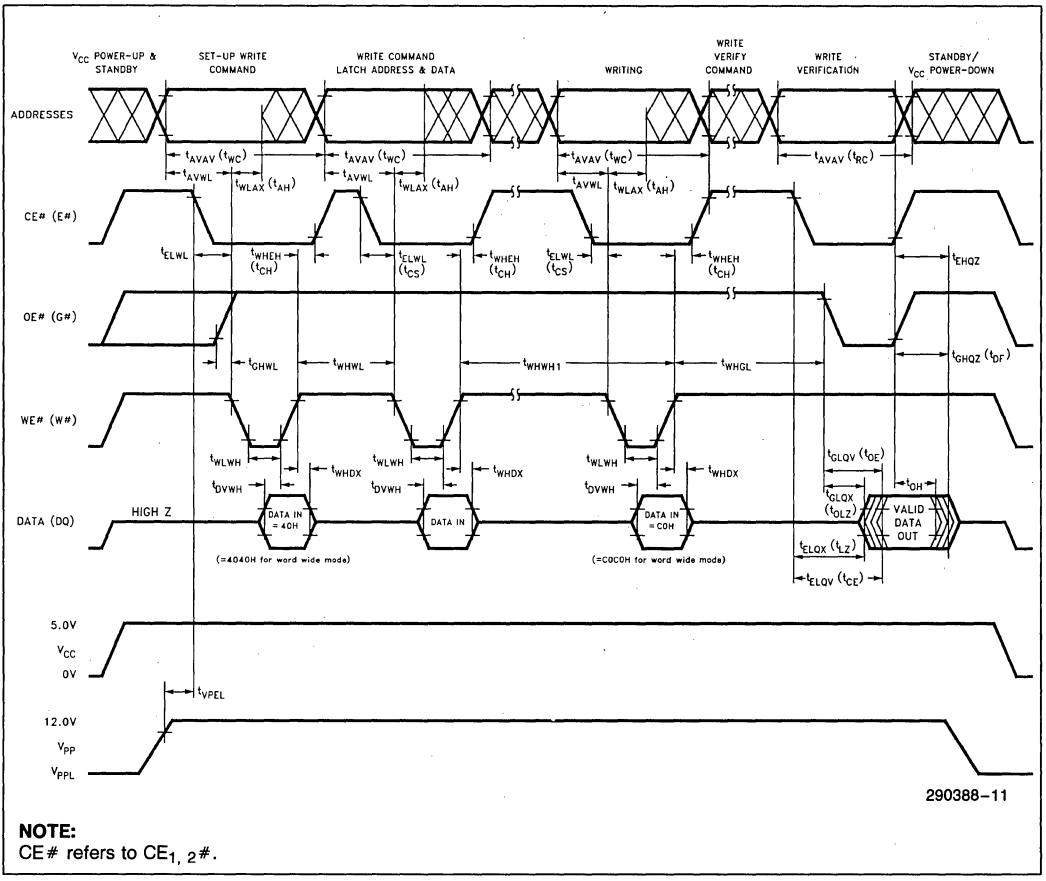
Symbol	Characteristic	Notes	Min	Max	Unit
t_{AVAV}/t_{WC}	Write Cycle Time	1, 2	200		ns
t_{AVWL}/t_{AS}	Address Set-up Time	1, 2	0		ns
t_{WLAX}/t_{AH}	Address Hold Time	1, 2	100		ns
t_{DVWH}/t_{DS}	Data Set-up Time	1, 2	80		ns
t_{WHDX}/t_{DH}	Data Hold Time	1, 2	30		ns
t_{WHGL}	Write Recovery Time before Read	1, 2	6		μs
t_{GHWL}	Read Recovery Time before Write	1, 2	0		μs
t_{WLOZ}	Output High-Z from Write Enable	1, 2	5		ns
t_{WHOZ}	Output Low-Z from Write Enable	1, 2		60	ns
t_{ELWL}/t_{CS}	Chip Enable Set-up Time before Write	1, 2	40		ns
t_{WHEH}/t_{CH}	Chip Enable Hold Time	1, 2	0		ns
t_{WLWH}/t_{WP}	Write Pulse Width	1, 2	100		ns
t_{WHWL}/t_{WPH}	Write Pulse Width High	1, 2	20		ns
t_{WHWH1}	Duration of Write Operation	1, 2, 3	10		μs
t_{WHWH2}	Duration of Erase Operation	1, 2, 3	9.5		ms
t_{VPEL}	V_{PP} Set-up Time to Chip Enable Low	1, 2	100		ns

NOTES:
1. Read timing parameters during read/write operations are the same as during read-only operations. Refer to AC Characteristics for Read-Only Operations.
2. Rise/Fall time ≤ 10 ns.
3. The integrated stop timer terminates the write/erase operations, thereby eliminating the need for a maximum specification.

ERASE/WRITE PERFORMANCE

Parameter	Notes	Min	Typ	Max	Unit
Zone Erase Time	1, 3, 4		2.0	30	sec
Zone Write Time	1, 2, 4		4.0	25	sec
MTBF	5		10(6)		Hrs

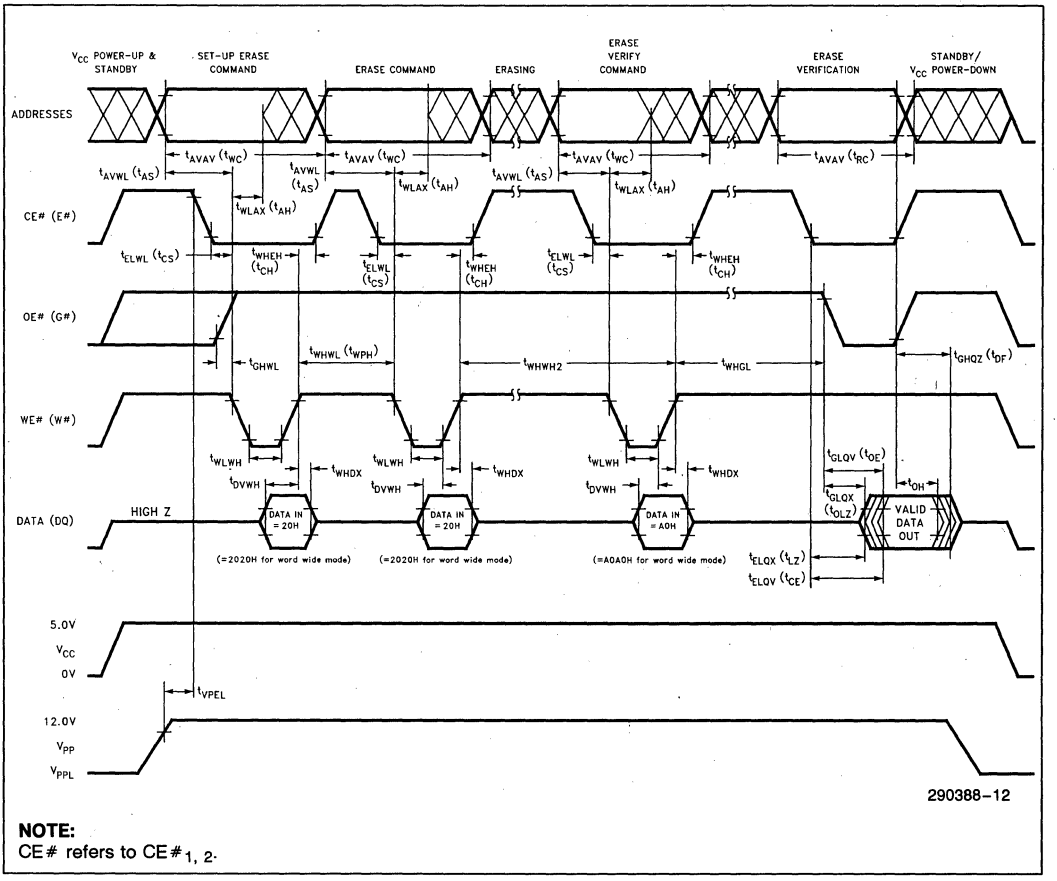
NOTES:
1. 25°C, 12.0V V_{PP}.
2. Minimum byte writing time excluding system overhead is 16 μs (10 μs program + 6 μs write recovery), while maximum is 400 μs/byte (16 μs x 25 loops allowed by algorithm). Max chip write time is specified lower than the worst case allowed by the write algorithm since most bytes write significantly faster than the worst case byte.
3. Excludes 00H writing Prior to Erasure.
4. One zone equals 256 Kbytes.
5. MTBF – Mean Time between Failure, 50% failure point for disk drives.



NOTE:
CE# refers to $CE_{1,2}$#.

Figure 11. AC Waveforms for Write Operations

6



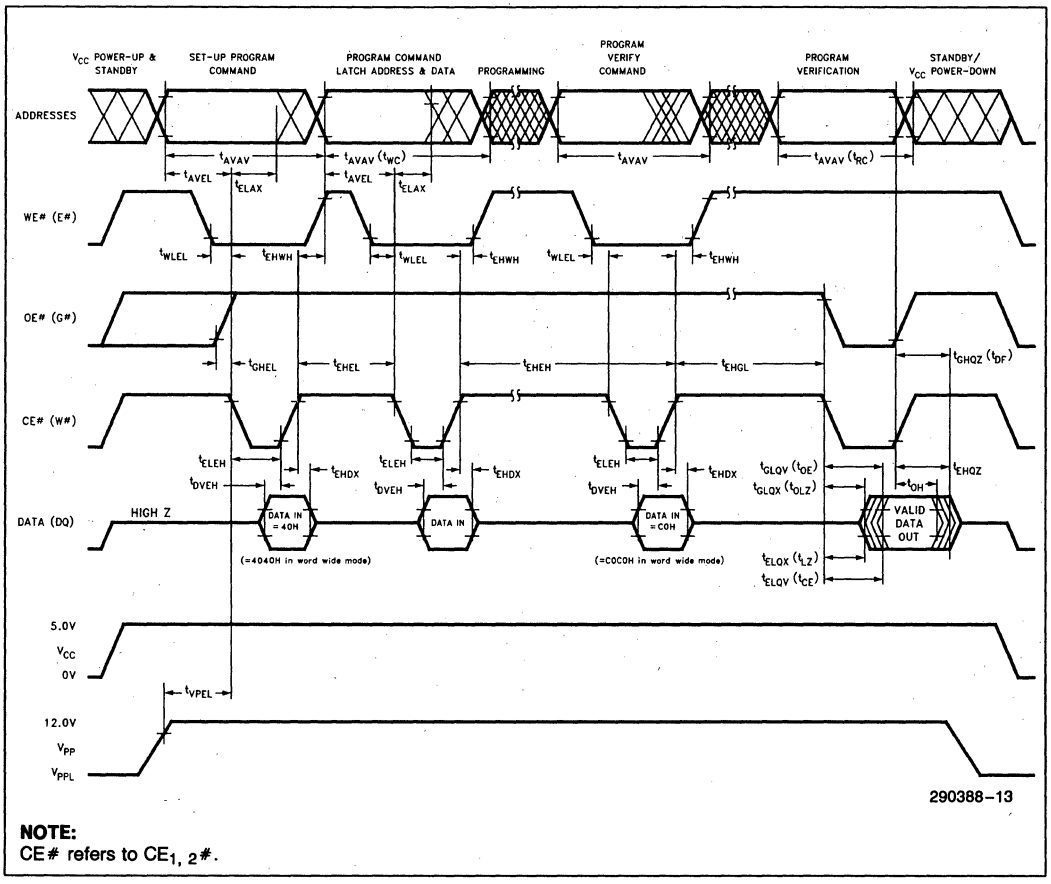
NOTE:
CE# refers to $CE#_{1, 2}$.

Figure 12. AC Waveforms for Erase Operations

ALTERNATIVE CE#-CONTROLLED WRITES

Symbol	Characteristic	Notes	Min	Max	Unit
t_{AVAV}	Write Cycle Time		200		ns
t_{AVEL}	Address Set-up Time		0		ns
t_{ELAX}	Address Hold Time		100		ns
t_{DVEH}	Data Set-up Time		80		ns
t_{EHDX}	Data Hold Time		30		ns
t_{EHGL}	Write Recovery Time before Read		6		µs
t_{GHEL}	Read Recovery Time before Write		0		µs
t_{WLEL}	Write Enable Set-Up Time before Chip-Enable		0		ns
t_{EHWH}	Write Enable Hold Time		0		ns
t_{ELEH}	Write Pulse Width	1	100		ns
t_{EHEL}	Write Pulse Width High		20		ns
t_{PEL}	V_{PP} Set-up Time to Chip Enable Low		100		ns

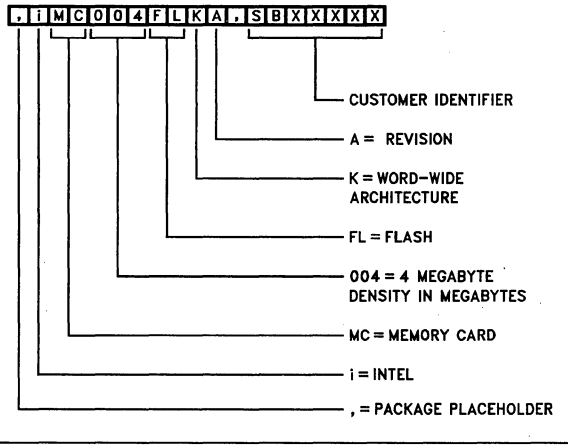
NOTES:
1. Chip Enable Controlled Writes: Write operations are driven by the valid combination of Chip Enable and Write Enable. In systems where Chip Enable defines the write pulse width (with a longer Write Enable timing waveform) all set-up, hold and inactive Write Enable times should be measured relative to the Chip Enable waveform.



NOTE:
CE# refers to $CE_{1,2}$#.

Figure 13. Alternate AC Waveforms for Write Operations

ORDERING INFORMATION



ADDITIONAL INFORMATION	ORDER NUMBER
ER-20, "ETOX II Flash Memory Technology"	294005
RR-60, "ETOX II Flash Memory Reliability Data Summary"	293002
AP-343, "Solutions for High Density Applications using Flash Memory"	292079
RR-70, "Flash Memory Card Reliability Data Summary"	293007

REVISION HISTORY

Number	Description
03	Removed PRELIMINARY Removed ExCA Compliance Section Clarified need for Valid Address during Commands Corrected $V_{PP} = V_{PPH}$ in Erase Algorithm Increased I_{CC2}–I_{CC5} D.C. Current Specifications for both Byte-Wide and Word-Wide modes. Revised and updated Application Section discussion Changed order number
04	Change signals with "_" to "#" Change T/C values

6

iMC002FLKA
2-MBYTE FLASH MEMORY CARD

- **Inherent Nonvolatility (Zero Retention Power)**
  - **— No Batteries Required for Back-up**
- **High-Performance Read**
  - **— 200 ns Maximum Access Time**
- **CMOS Low Power Consumption**
  - **— 25 mA Typical Active Current (X8)**
  - **— 400 μA Typical Standby Current**
- **Flash Electrical Zone-Erase**
  - **— 2 Seconds Typical per 256 Kbyte Zone**
  - **— Multiple Zone-Erase**
- **Random Writes to Erased Zones**
  - **— 10 μs Typical Byte Write**
- **Write Protect Switch to Prevent Accidental Data Loss**
- **Command Register Architecture for Microprocesssor/Microcontroller Compatible Write Interface**
- **ETOX™ II Flash Memory Technology**
  - **— 5V Read, 12V Erase/Write**
  - **— High-Volume Manufacturing Experience**
- **PCMCIA/JEIDA 68-Pin Standard**
  - **— Byte- or Word-wide Selectable**
- **Independent Software & Hardware Vendor Support**
  - **— Integrated System Solution Using Flash Filing Systems**

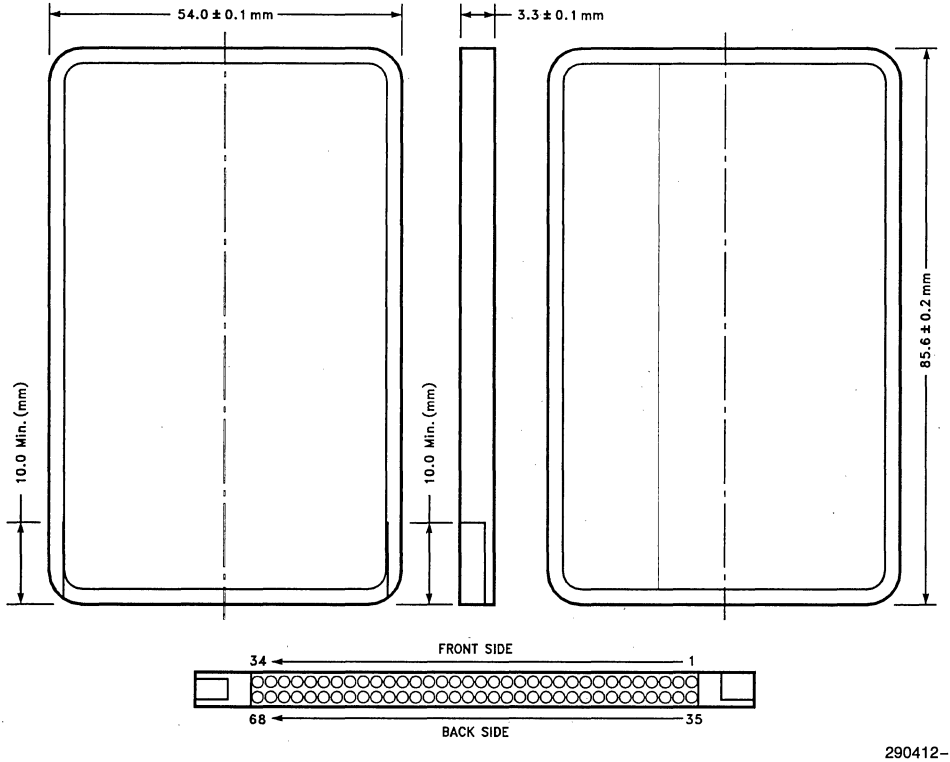
Intel's iMC002FLKA Flash Memory Card is the removable solution for storing and transporting important user data and application code. The combination of rewritability and nonvolatility make the Intel Flash Memory Card ideal for data acquisition and updatable firmware applications. Designing with Intel's Flash Memory Card enables OEM system manufacturers to produce portable and dedicated function systems that are higher performance, more rugged, and consume less power.

The iMC002FLKA conforms to the PCMCIA 1.0 international standard, providing standardization at the hardware and data interchange level. OEMs may opt to write the Card Information Structure (CIS) at the memory card's address 00000H with a format utility. This information provides data interchange functional capability. The 200 ns access time allows for "execute-in-place" capability, for popular low-power microprocessors. Intel's 2-MByte Flash Memory Card operates in a byte-wide and word-wide configuration providing performance/power options for different systems.

Intel's Flash Memory card employs Intel's ETOX II Flash Memories. Filing systems, such as Microsoft's* Flash File System (FFS), facilitate data file storage and card erasure using a purely nonvolatile medium in the DOS environment. Flash filing systems, coupled with the Intel Flash Memory Card, effectively create an all-silicon nonvolatile read/write random access memory system that is more reliable and higher performance than disk-based memory systems.

*Microsoft is a trademark of Microsoft Corp.

March 1994
Order Number: 290412-003



290412-1

1	GND	18	V_{PP1}	35	GND	52	V_{PP2}
2	D_3	19	A_{16}	36	CD_1#	53	NC
3	D_4	20	A_{15}	37	D_{11}	54	NC
4	D_5	21	A_{12}	38	D_{12}	55	NC
5	D_6	22	A_7	39	D_{13}	56	NC
6	D_7	23	A_6	40	D_{14}	57	NC
7	CE_1#	24	A_5	41	D_{15}	58	NC
8	A_{10}	25	A_4	42	CE_2#	59	NC
9	OE#	26	A_3	43	NC	60	NC
10	A_{11}	27	A_2	44	NC	61	REG#
11	A_9	28	A_1	45	NC	62	BVD_2#[2]
12	A_8	29	A_0	46	A_{17}	63	BVD_1#[2]
13	A_{13}	30	D_0	47	A_{18}	64	D_8
14	A_{14}	31	D_1	48	A_{19}	65	D_9
15	WE#	32	D_2	49	A_{20}	66	D_{10}
16	NC	33	WP	50	NC	67	CD_2#
17	V_{CC}	34	GND	51	V_{CC}	68	GND

NOTES:
1. REG# = register memory select = No Connect (NC), unused. When REG# is brought low, PCMCIA/JEIDA standard card information structure data is expected. This is accomplished by formatting the card with this data.
2. BVD# = battery detect voltage = Pulled High through Pull-Up Resistor.

Figure 1. iMC002FLKA Pin Configurations

Table 1. Pin Description

Symbol	Type	Name and Function
A_0–A_{20}	I	**ADDRESS INPUTS** for memory locations. Addresses are internally latched during a write cycle.
D_0–D_{15}	I/O	**DATA INPUT/OUTPUT:** Inputs data during memory write cycles; outputs data during memory read cycles. The data pins are active high and float to tri-state OFF when the card is deselected or the outputs are disabled. Data is internally latched during a write cycle.
CE_1#, CE_2#	I	**CARD ENABLE:** Activates the card's high and low byte control logic, input buffers, zone decoders, and associated memory devices. CE# is active low; CE# high deselects the memory card and reduces power consumption to standby levels.
OE#	I	**OUTPUT ENABLE:** Gates the cards output through the data buffers during a read cycle. OE# is active low.
WE#	I	**WRITE ENABLE** controls writes to the control register and the array. Write enable is active low. Addresses are latched on the falling edge and data is latched on the rising edge of the WE# pulse. **NOTE:** With $V_{PP} \leq 6.5V$, memory contents cannot be altered.
V_{PP1}, V_{PP2}		**ERASE/WRITE POWER SUPPLY** for writing the command register, erasing the entire array, or writing bytes in the array.
V_{CC}		**DEVICE POWER SUPPLY** (5V ±5%).
GND		**GROUND**
CD_1#, CD_2#	O	**CARD DETECT.** The card is detected at $CD_{1/2}$# = ground.
WP	O	**WRITE PROTECT.** All write operations are disabled with WP = active high.
NC		**NO INTERNAL CONNECTION** to device. Pin may be driven or left floating.
BVD_1#, BVD_2#	O	**BATTERY VOLTAGE DETECT. NOT REQUIRED.**

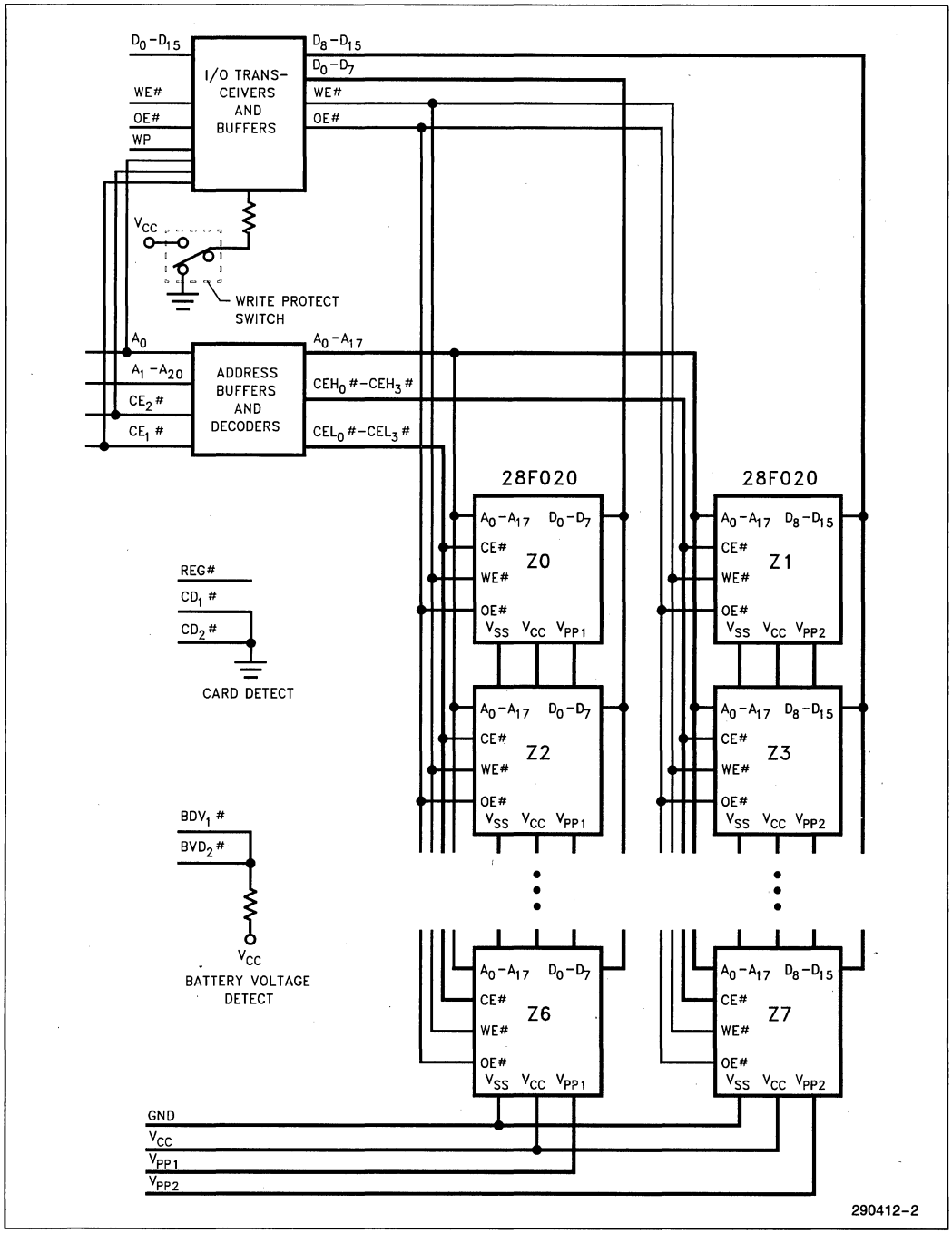


Figure 2. iMC002FLKA Block Diagram

6

APPLICATIONS

The iMC002FLKA Flash Memory Card allows for the storage of data and application programs on a purely solid-state removable medium. System resident flash filing systems, such as Microsoft's Flash File System, allow Intel's ETOX II highly reliable Flash Memory Card to effectively function as a physical disk drive.

User application software stored on the flash memory card substantially reduces the slow disk-DRAM download process. Replacing the disk results in a dramatic enhancement of read performance and substantial reduction of active power consumption, size, and weight—considerations particularly important in portables and dedicated systems. The iMC002FLKA's high performance read access time and command register microprocessor write interface allows for use of the flash memory system in an "execute-in-place" architecture. This configuration eliminates the need for the redundancy associated with DRAM and Disk memory system architectures. ROM based operating systems, such as Microsoft's MS-DOS ROM Version allow for "instant-on" capability. This enables the design of systems that boot, operate, store data files, and execute application code from/to purely nonvolatile memory.

The PCMCIA/JEIDA 68-pin interface with flash filing systems enables the end-user to transport data and application code between portables and host systems. Intel Flash Memory Cards provide durable nonvolatile memory storage protecting valuable user code and data.

For systems currently using a static RAM/battery configuration for data acquisition, the iMC002FLKA's inherent nonvolatility eliminates the need for battery backup. The concern of battery failure no longer exists, an important consideration for portable computers and medical instruments, both requiring continuous operation. The iMC002FLKA consumes no power when the system is off. In addition, the iMC002FLKA offers a considerable cost and density advantage over memory cards based on static RAM with battery backup.

The flash memory card's electrical zone-erasure, byte writability, and complete nonvolatility fit well with data accumulation and recording needs. Electrical zone-erasure gives the designer the flexibility to selectively rewrite zones of data while saving other zones for infrequently updated look-up tables, for example.

PRINCIPLES OF OPERATION

Intel's Flash Memory Card combines the functionality of two mainstream memory technologies: the rewritability of RAM and the nonvolatility of EPROM. The flash memory card consists of an array of individual memory devices, each of which defines a physical zone. The iMC002FLKA's memory devices erase as individual blocks, equivalent in size to the 256 Kbyte zone. Multiple zones can be erased simultaneously provided sufficient current for the appropriate number of zones (memory devices). Note, multiple zone erasure requires higher current from both the V_{PP} and V_{CC} power supplies. Erased zones can then be written in bit- or byte-at-a-time fashion and read randomly like RAM. Bit level write capability also supports disk emulation.

In the absence of high voltage on the $V_{PP1/2}$ pins, the iMC002FLKA remains in the read-only mode. Manipulation of the external memory card-control pin yields the standard read, standby, and output disable operations.

The same read, standby, and output disable operations are available when high voltage is applied to the $V_{PP1/2}$ pins. In addition, high voltage on $V_{PP1/2}$

enables erasure and rewriting of the accessed zone(s). All functions associated with altering zone contents—erase, erase verify, write, and write verify—are accessed via the command register.

Commands are written to the internal memory register(s), decoded by zone size, using standard microprocessor write timings. Register contents for a given zone serve as input to that zone's internal state-machine which controls the erase and rewrite circuitry. Write cycles also internally latch addresses and data needed for write and erase operations. With the appropriate command written to the register(s), standard microprocessor read timings output zone data, or output data for erase and write verification.

Byte-wide or Word-wide Selection

The flash memory card can be read, erased, and written in a byte-wide or word-wide mode. In the word-wide configuration V_{PP1} and/or CE_1# control the LO-Byte while V_{PP2} and CE_2# control the HI-Byte (A_0 = don't care).

Read, Write, and Verify operations are byte- or word-oriented, thus zone independent. Erase Setup and Begin Erase Commands are zone dependent such that commands written to any address within a 256 Kbyte zone boundary initiate the erase operation in that zone (or two 256 Kbyte zones under word-wide operation).

Conventional x8 operation uses CE_1# active-low, with CE_2# high, to read or write data through the D_0–D_7 only. "Even bytes" are accessed when A_0 is low, corresponding to the low byte of the complete x16 word. When A_0 is high, the "odd byte" is accessed by transposing the high byte of the complete x16 word onto the D_0–D_7 outputs. This odd byte corresponds to data presented on D_8–D_{15} pins in x16 mode.

Note that two zones logically adjacent in x16 mode are multiplexed through D_0–D_7 in x8 mode and are toggled by the A_0 address. Thus, zone specific erase operations must be kept discrete in x8 mode by addressing even bytes only for one-half of the zone pair, then addressing odd bytes only for the other half.

Card Detection

The flash memory card features two card detect pins ($CD_{1/2}$#) that allow the host system to determine if the card is properly loaded. Note that the two pins are located at opposite ends of the card. Each CD# output should be read through a port bit. Should only one of the two bits show the card to be present, then the system should instruct the user to re-insert the card squarely into the socket. Card detection can also tell the system whether or not to redirect drives in the case of system booting. $CD_{1/2}$# is active low, internally tied to ground.

Write Protection

The flash memory card features three types of write protection. The first type features a mechanical Write Protect Switch that disables the circuitry that control Write Enable to the flash devices. When the switch is activated, the WE# internal to the cord is forced high, which disables any writes to the Command Register. The second type of write protection is based on the PCMCIA/JEIDA socket. Unique pin length assignments provide protective power supply sequencing during hot insertion and removal. The third type operates via software control through the Command Register when the card resides in its connector. The Command Register of each zone is only active when $V_{PP1/2}$ is at high voltage. Depending upon the application, the system designer may choose to make $V_{PP1/2}$ power supply switchable—available only when writes are desired. When $V_{PP1/2}$ = V_{PPL}, the contents of the register default to the read command, making the iMC002FLKA a read-only memory card. In this mode, the memory contents cannot be altered.

The system designer may choose to leave $V_{PP1/2}$ = V_{PPH}, making the high voltage supply constantly available. In this case, all Command Register functions are inhibited whenever V_{CC} is below the write lockout voltage, V_{LKO}. (See the section on Power Up/Down Protection.) The iMC002FLKA is designed to accommodate either design practice, and to encourage optimization of the processor-memory card interface.

BUS OPERATIONS

Read

The iMC002FLKA has two control functions, both of which must be logically active, to obtain data at the outputs. Card Enable (CE#) is the power control and should be used for high and/or low zone(s) selection. Output Enable (OE#) is the output control and should be used to gate data from the output pins, independent of accessed zone selection. In the byte-wide configuration, only one CE# is required. The word-wide configuration requires both CE#s active low.

When $V_{PP1/2}$ is high (V_{PPH}), the read operations can be used to access zone data and to access data for write/erase verification. When $V_{PP1/2}$ is low (V_{PPL}), only read accesses to the zone data are allowed.

Output Disable

With Output Enable at a logic-high level (V_{IH}), output from the card is disabled. Output pins are placed in a high-impedance state.

Standby

With one Card Enable at a logic-high level, the standby operation disables one-half of the x16 output's read/write buffer. Further, only the zone corresponding to the selected address within the upper or lower $CE_{1/2}$# bank is active at a time. (NOTE: A_0 must be low to select the low half of the x16 word when CE_2# = 1 and CE_1# = 0.) All other zones are deselected, substantially reducing card power consumption. For deselected banks, the outputs are placed in a high-impedance state, independent of the Output Enable signal. If the iMC002FLKA is deselected during erasure, writing, or write/erase verification, the accessed zone draws active current until the operation is terminated.

Intelligent Identifier Command

The manufacturer- and device-codes can be read via the Command Register, for instances where the iMC002FLKA is erased and rewritten in a universal reader/writer. Following a write of 90H to a zone's Command Register, a read from address location 00000H on any zone outputs the manufacturer code (89H). A read from address 0002H outputs the memory device code (BDH).

Write

Zone erasure and rewriting are accomplished via the Command Register, when high voltage is applied to $V_{PP1/2}$. The contents of the register serve as input to that zone's internal state-machine. The state-machine outputs dictate the function of the targeted zone.

The Command Register itself does not occupy an addressable memory location. The register is a latch used to store the command, along with address and data information needed to execute the command.

The Command Register is written by bringing Write Enable to a logic-low level (V_{IL}), while Card Enable(s) is/are low. Addresses are latched on the falling edge of Write Enable, while data is latched on the rising edge of the Write Enable pulse. Standard microprocessor write timings are used.

Refer to AC Write Charcteristics and the Erase/Write Waveforms for specific timing parameters.

COMMAND DEFINITIONS

When low voltage is applied to the V_{PP} pin(s), the contents of the zone Command Register(s) default to 00H, enabling read-only operations.

Placing high voltage on the V_{PP} pin(s) enable(s) read/write operations. Zone operations are selected by writing specific data patterns into the Command Register. Tables 3 and 4 define these iMC002FLKA register commands for both byte-wide and word-wide configurations.

All commands written to the Command Register require that the zone address be valid or the incorrect zone will receive the command. Any Command/Data Write or Data Read requires the correct valid address.

Table 2. Bus Operations

Pins / Operation		Notes	[1, 7] V_{PP2}	[1, 7] V_{PP1}	A0	CE_2#	CE_1#	OE#	WE#	D_8–D_{15}	D_0–D_7
READ-ONLY	Read (x8)	8	V_{PPL}	V_{PPL}	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Even
	Read (x8)	9	V_{PPL}	V_{PPL}	V_{IH}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Odd
	Read (x8)	10	V_{PPL}	V_{PPL}	X	V_{IL}	V_{IH}	V_{IL}	V_{IH}	Data Out	Tri-state
	Read (x16)	11	V_{PPL}	V_{PPL}	X	V_{IL}	V_{IL}	V_{IL}	V_{IH}	Data Out	Data Out
	Output Disable		V_{PPL}	V_{PPL}	X	X	X	V_{IH}	V_{IH}	Tri-state	Tri-state
	Standby		V_{PPL}	V_{PPL}	X	V_{IH}	V_{IH}	X	X	Tri-state	Tri-state
READ/WRITE	Read (x8)	3, 8	V_{PPX}	V_{PPH}	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Even
	Read (x8)	3, 9	V_{PPH}	V_{PPX}	V_{IH}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Odd
	Read (x8)	10	V_{PPH}	V_{PPX}	X	V_{IL}	V_{IH}	V_{IL}	V_{IH}	Data Out	Tri-state
	Read (x16)	3, 11	V_{PPH}	V_{PPH}	X	V_{IL}	V_{IL}	V_{IL}	V_{IH}	Data Out	Data Out
	Write (x8)	5, 8	V_{PPX}	V_{PPH}	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{IL}	Tri-state	Data In-Even
	Write (x8)	9	V_{PPH}	V_{PPX}	V_{IH}	V_{IH}	V_{IL}	V_{IH}	V_{IL}	Tri-state	Data In-Odd
	Write (x8)	10	V_{PPH}	V_{PPX}	X	V_{IL}	V_{IH}	V_{IH}	V_{IL}	Data In	Tri-state
	Write (x16)	11	V_{PPH}	V_{PPH}	X	V_{IL}	V_{IL}	V_{IH}	V_{IL}	Data In	Data In
	Standby	4	V_{PPH}	V_{PPH}	X	V_{IH}	V_{IH}	X	X	Tri-state	Tri-state
	Output Disable		V_{PPH}	V_{PPH}	X	X	X	V_{IH}	V_{IL}	Tri-state	Tri-state

NOTES:
1. Refer to DC Characteristics. When $V_{PP1/2} = V_{PPL}$ memory contents can be read but not written or erased.
2. Manufacturer and device codes may be accessed via a command register write sequence. Refer to Table 3. All other addresses low.
3. Read operations with $V_{PP1/2} = V_{PPH}$ may access array data or the Intelligent Identifier codes.
4. With $V_{PP1/2}$ at high voltage, the standby current equals $I_{CC} + I_{PP}$ (standby).
5. Refer to Table 3 for valid Data-In during a write operation.
6. X can be V_{IL} or V_{IH}.
7. $V_{PPX} = V_{PPH}$ or V_{PPL}.
8. This x8 operation reads or writes the low byte of the x16 word on DQ_{0-7}, i.e., A_0 low reads "even" byte in x8 mode.
9. This x8 operation reads or writes the high byte of the x16 word on DQ_{0-7} (transposed from DQ_{8-15}), i.e., A_0 high reads "odd" byte in x8 mode.
10. This x8 operation reads or writes the high byte of the x16 on DQ_{8-15}. A_0 is "don't care."
11. A_0 is "don't care," unused in x16 mode. High and low bytes are presented simultaneously.

Table 3. Command Definitions Byte-Wide Mode

Command	Notes	Bus Cycles Req'd	First Bus Cycle			Second Bus Cycle		
			Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory	2	1	Write	RA	00H			
Read Intelligent ID Codes	4	3	Write	IA	90HT	Read		
Set-up Erase/Erase	5	2	Write	ZA	20H	Write	ZA	20H
Erase Verify	5	2	Write	EA	A0H	Read	EA	EVD
Set-up Write/Write	6	2	Write	WA	40H	Write	WA	WD
Write Verify	6	2	Write	WA	C0H	Read	WA	WVD
Reset	2, 7, 8	2	Write	ZA	FFH	Write	ZA	FFH

Table 4. Command Definitions Word-Wide Mode

Command	Notes	Bus Cycles Req'd	First Bus Cycle			Second Bus Cycle		
			Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory	2	1	Write	RA	0000H			
Read Intelligent ID Codes	4	3	Write	IA	9090H	Read		
Set-up Erase/Erase	5	2	Write	ZA	2020H	Write	ZA	2020H
Erase Verify	5	2	Write	EA	A0A0H	Read	EA	EVD
Set-up Write/Write	6	2	Write	WA	4040H	Write	WA	WD
Write Verify	6	2	Write	WA	C0C0H	Read	WA	WVD
Reset	2, 7, 8	2	Write	ZA	FFFFH	Write	ZA	FFFFH

NOTES:
1. Bus operations are defined in Table 2.
2. IA = Identifier address: 00H for manufacturer code, 01H for device code.
   EA = Address of memory location to be read during erase verify.
   RA = Read Address
   WA = Address of memory location to be written.
   ZA = Address of 256 Kbyte zones involved in erase or Reset operations.
   Addresses are latched on the falling edge of the Write Enable pulse.
3. ID = Data read from location IA during device identification. (Mfr = 89H, Device = BDH).
   EVD = Data read from location EA during erase verify.
   WD = Data to be written at location WA. Data is latched on the rising edge of Write Enable.
   WVD = Data read from location WA during write verify. WA is latched on the Write command.
4. Following the Read Intelligent ID command, two read operations access manufacturer and device codes.
5. Figure 5 illustrates the Erase Algorithm.
6. Figure 6 illustrates the Write Algorithm.
7. The second bus cycle must be followed by the desired command register write.
8. The Reset command operates on a zone basis. To reset the entire card requires reset write cycles to each zone.

Read Command

While $V_{PP1/2}$ is high, for erasure and writing, zone memory contents can be accessed via the read command. The read operation is initiated by writing 00H (0000H for the word-wide configuration) into the zone Command Register(s). Microprocessor read cycles retrieve zone data. The accessed zone remains enabled for reads until the Command Register(s) contents are altered.

The default contents of each zone's register(s) upon $V_{PP1/2}$ power-up is 00H (00000H for word-wide). This default value ensures that no spurious alteration of memory card contents occurs during the $V_{PP1/2}$ power transition. Where the $V_{PP1/2}$ supply is left at V_{PPH}, the memory card powers-up and remains enabled for reads until the command Register contents of targeted zones are changed. Refer to the AC Read Characteristics and Waveforms for specific timing parameters.

Intelligent Identifier Command

Each zone of the iMC002FLKA contains an Intelligent Identifier to identify memory card device characteristics. The operation is initiated by writing 90H (9090H for word-wide) into the Command Register(s) with zone address. Following the command write, a read cycle from address 00000H retrieves the manufacturer code 89H (8989H for word-wide). A read cycle from address 0002H returns the device code BDH (BDBDH for word-wide). To terminate the operation, it is necessary to write another valid command into the register(s).

Set-up Erase/Erase Commands

Set-up Erase stages the targeted zone(s) for electrical erasure of all bytes in the zone. The set-up erase operation is performed by writing 20H to the Command Register (2020H for word-wide) with zone address.

To commence zone-erasure, the erase command (20H or 2020H) must again be written to the register(s) with zone address. The erase operation begins with the rising edge of the Write-Enable pulse and terminates with the rising edge of the next Write-Enable pulse (i.e., Erase-Verify Command with zone address).

This two-step sequence of set-up followed by execution ensures that zone memory contents are not accidentally erased. Also, zone-erasure can only occur when high voltage is applied to the $V_{PP1/2}$ pins. In the absence of this high voltage, zone memory contents are protected against erasure. Refer to AC Erase Characterstics and Waveforms for specific timing parameters.

Erase-Verify Command

The erase command erases all of the bytes of the zone in parallel. After each erase operation, all bytes in the zone must be individually verified. In byte-mode operations, zones are segregated by A_0 in odd and even banks; erase and erase verify operations must be done in complete passes of even-bytes-only then odd-bytes-only. See the Erase Algorithm for byte-wide mode. The erase verify operation is initiated by writing A0H (A0A0H for word-wide) into the Command Register(s). The address for the byte(s) to be verified must be supplied as it is latched on the falling edge of the Write Enable pulse. The register write terminates the erase operation with the rising edge of its Write Enable pulse.

The enabled zone applies an internally-generated margin voltage to the addressed byte. Reading FFH from the addressed byte indicates that all bits in the byte are erased. Similarly, reading FFFFH from the addressed word indicates that all bits in the word are erased.

The erase-verify command must be written to the Command Register prior to each byte (word) verification to latch its address. The process continues for each byte (word) in the zone(s) until a byte (word) does not return FFH (FFFFH) data, or the last address is accessed.

In the case where the data read is not FFH (FFFFH), another erase operation is performed. (Refer to Set-up Erase/Erase.) Verification then resumes from the address of the last-verified byte (word). Once all bytes (words) in the zone(s) have been verified, the erase step is complete. The accessed zone can now be written. At this point, the verify operation is terminated by writing a valid command (e.g., Write Set-up) to the Command Register. The Erase algorithms for byte-wide and word-wide configurations illustrate how commands and bus operations are combined to perform electrical erasure of the iMCC002FLKA. Refer to AC Erase Characteristics and Waveforms for specific timing parameters.

Set-up Write/Write Commands

Set-up write is a command-only operation that stages the targeted zone for byte writing. Writing 40H (4040H) into the Command Register(s) performs the set-up operation.

Once the write set-up operation is performed, the next Write Enable pulse causes a transition to an active write operation. Addresses are internally latched on the falling edge of the Write Enable pulse. Data is internally latched on the rising edge of the Write Enable pulse. The rising edge of Write Enable also begins the write operation. The write operation terminates with the next rising edge of Write Enable, which is used to write the verify command. Refer to AC Write Characteristics and Waveforms for specific timing parameters.

Write Verify Command

The iMC002FLKA is written on a byte-by-byte or word-by-word basis. Byte or word writing may occur sequentially or at random. Following each write operation, the byte or word just written must be verified.

The write-verify operation is initiated by writing C0H (C0C0H) into the Command Register(s) with correct address. The register write(s) terminate(s) the write operation with the rising edge of its Write Enable pulse. The write-verify operation stages the accessed zone(s) for verification of the byte or word last written. The zone(s) apply(ies) an internally-generated margin voltage to the byte or word. A microprocessor read cycle outputs the data. A successful comparison between the written byte or word and true data means that the byte or word is successfully written. The write operation then proceeds to the next desired byte or word location. The Write algorithms for byte-wide and word-wide configurations illustrate how commands are combined with bus operations to perform byte and word writes. Refer to AC Write Characteristics and Waveforms for specific timing parameters.

Reset Command

A reset command is provided as a means to safely abort the erase- or write-command sequences. Following either set-up command (erase or write) with two consecutive writes of FFH (FFFFH for word-wide) will safely abort the operation. Zone memory contents will not be altered. A valid command must then be written to place the accessed zone in the desired state.

EXTENDED ERASE/WRITE CYCLING

Intel has designed extended cycling capability into its ETOX II flash memory technology enabling a flash memory card with a MTBF that is many times more reliable than rotating disk technology. Resulting improvements in cycling reliability come without increasing memory cell size or complexity. First, an advanced tunnel oxide increases the charge carrying ability ten-fold. Second, the oxide area per cell subjected to the tunneling electric field minimizes the probability of oxide defects in the region. The lower electric field greatly reduces oxide stress and the probability of failure.

WRITE ALGORITHMS

The write algorithm(s) use write operations of 10 μs duration. Each operation is followed by a byte or word verification to determine when the addressed byte or word has been successfully written. The algorithm(s) allows for up to 25 write operations per byte or word, although most bytes and words verify on the first or second operation. The entire sequence of writing and byte/word verification is performed with V_{PP} at high voltage.

ERASE ALGORITHM

The Erase algorithm(s) yield(s) fast and reliable electrical erasure of memory contents. The algorithm employs a closed-loop flow, similar to the write algorithm, to simultaneously remove charge from all bits in the accessed zone(s).

Erasure begins with a read of memory zone contents. Reading FFH (FFFFH) data from the accessed zone(s) can be immediately followed by writing to the desired zone(s).

For zones being erased and rewritten, uniform and reliable erasure is ensured by first writing all bits in the accessed zone to their charged state (data = 00H byte-wide, 00000H word-wide). This is accomplished, using the write algorithm, in approximately four seconds per zone.

Erase execution then continues with an initial erase operation. Erase verification (data = FFH byte-wide, FFFFH word-wide) begins at address 00000H and continues through the zone to the last address, or until data other than FFH (FFFFH) is encountered. (Note: byte-wide erase operation requires separate even- and odd-address passes to handle the individual 256 Kbyte zones.) With each erase operation, an increasing number of bytes or words verify to the erased state. Erase efficiency may be improved by storing the address of the last byte or word verified in a register(s). Following the next erase operation, verification starts at the stored address location. Follow this procedure until all bytes in the zone are erased. Then, re-start the procedure for the next zone or word-wide zone pair. Erasure typically occurs in two seconds per zone.

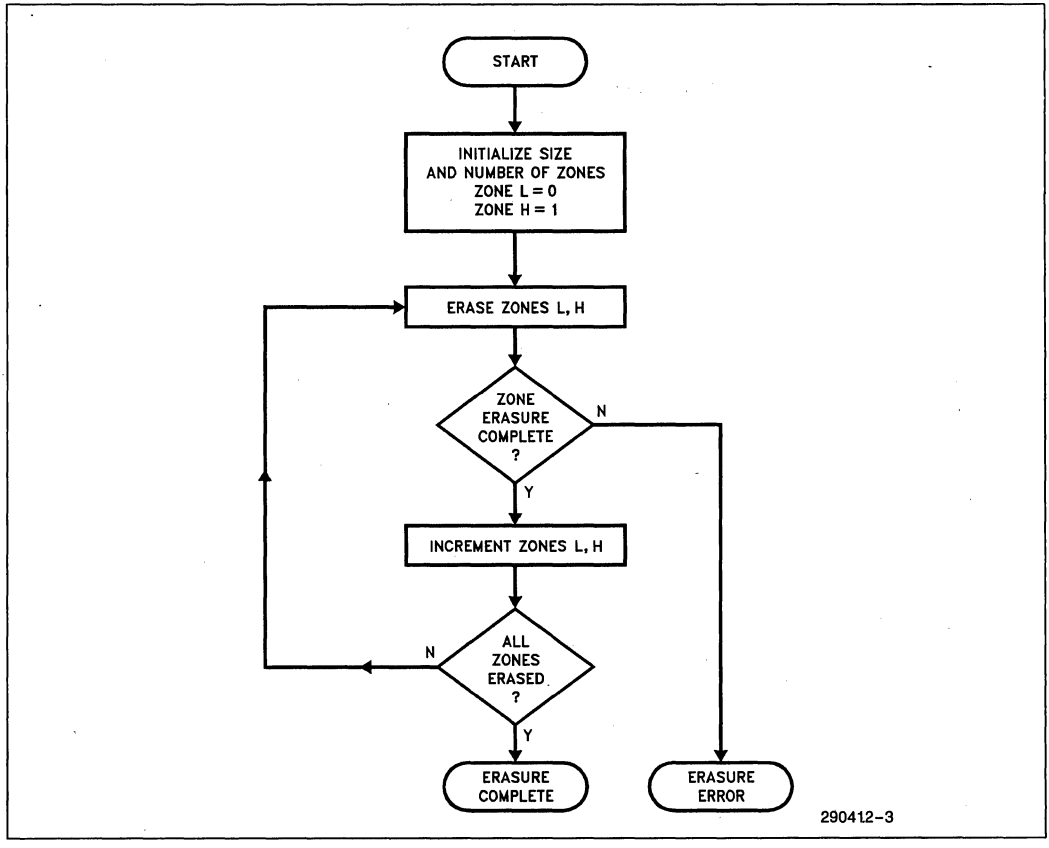
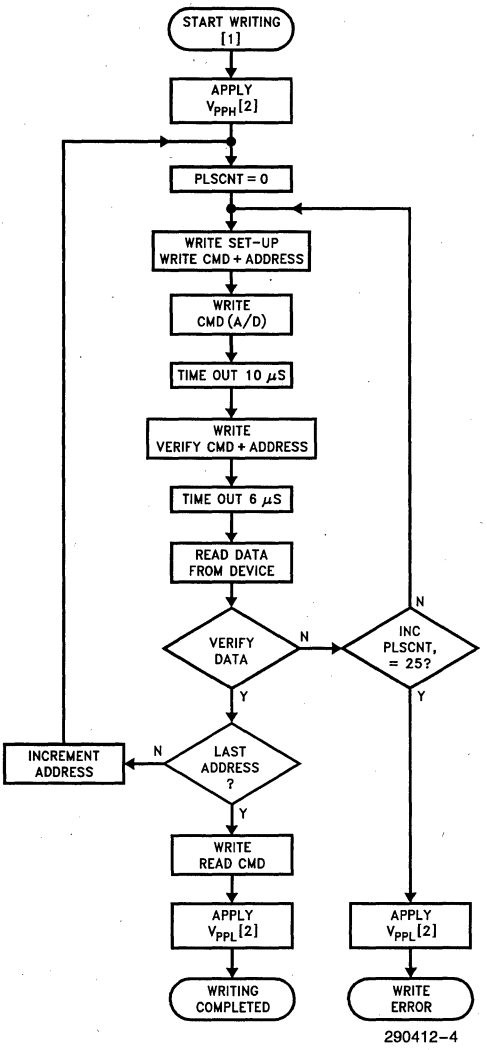


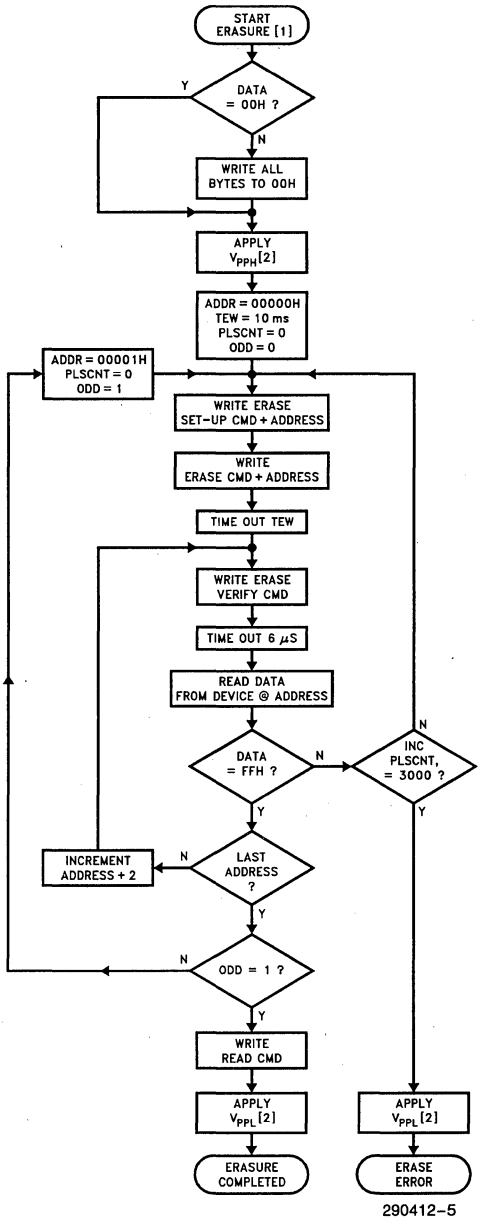
Figure 3. Full Card Erase Flow



Bus Operation	Command	Comments
Standby		Wait for V_{PP} ramp to V_{PPH} (= 12.0V) (2)
		Initialize pulse-count
Write	Set-up Write	Data = 40H + Valid Address
Write	Write	Valid address/data
Standby		Duration of Write operation (t_{WHWH1})
Write	Write(3) Verify	Data = C0H + Valid Address; Stops(4) Write Operation
Standby		t_{WHGL}
Read		Read byte to verify Write Operation at Valid Address
Standby		Compare data output to data expected
Write	Read	Data = 00H, resets the register for read operations.
Standby		Wait for V_{PP} ramp to V_{PPL}(2)

NOTES:
1. CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.
2. See DC Characteristics for the value of V_{PPH} and V_{PPL}.
3. Write Verify is only performed after a byte write operation. A final read/compare may be performed (optional) after the register is written with the Read command.
4. Refer to principles of operation.

Figure 4. Write Algorithm for Byte-Wide Mode



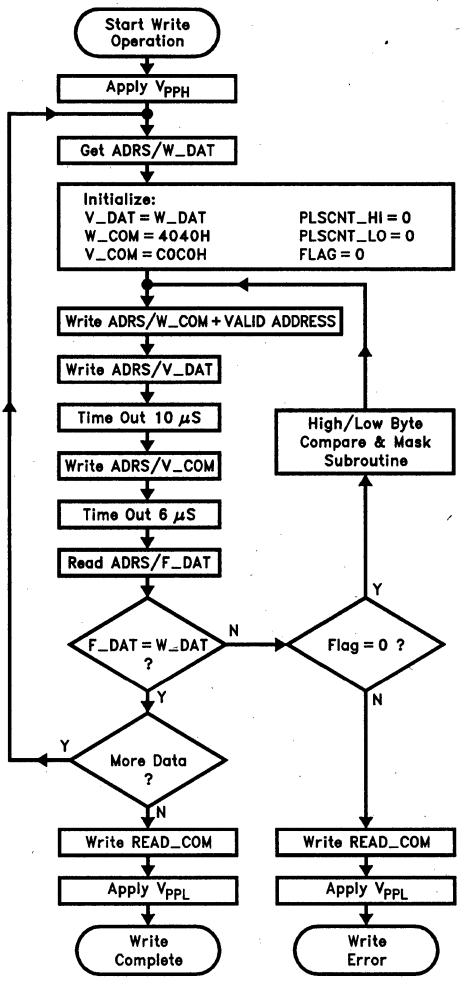
290412-5

Bus Operation	Command	Comments
Standby		Wait for V_{PP} ramp to V_{PPH} (= 12.0V)(2)
		Use with Write Operation Algorithm
		Initialize even/odd Addresses, Erase Pulse Width, and Pulse Count
Write	Set-up Erase	Data = 20H + Address
Write	Erase	Data = 20H + Address
Standby		Duration of Erase operation (t_{WHWH2})
Write	Erase Verify(3)	Addr = Byte to verify; Data = A0H; Stops
Standby		Erase Operation(4) t_{WHGL}
Read		Read byte to verify erasure at address
Standby		Compares output to FFH increment pulse count
Write	Read	Data = 00H, resets the register for read operations.
Standby		Wait for V_{PP} ramp to V_{PPL}(2)

NOTES:
1. CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.
2. See DC Characteristics for the value of V_{PPH} and V_{PPL}.
3. Erase Verify is only performed after a chip erasure. A final read/compare may be performed (optional) after the register is written with the Read command.
4. Refer to principles of operation.

Figure 5. Erase Algorithm for Byte-Wide Mode

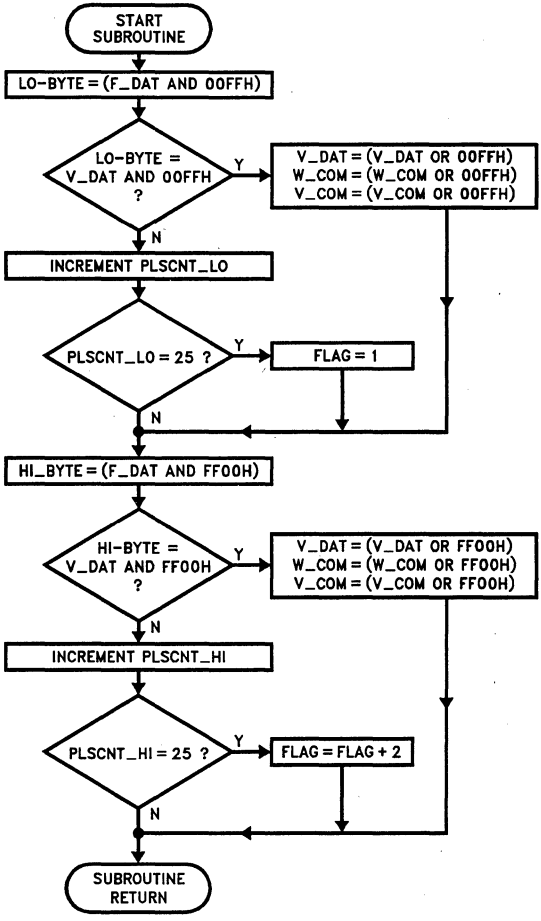
6



Comments
Wait for V_{PP} ramp to V_{PPH}
ADRS = address to write W_DAT = data word to write
Initialize Data Word Variables: V_DAT = valid data W_COM = Write Command V_COM = Write Verify Command PLSCNT_HI = HI Byte Pulse Counter PLSCNT_LO = LO Byte Pulse Counter FLAG = Write Error Flag Write Set-up Command Address needs to be Valid Write
See Write Verify and Mask Subroutine
Write Verify Command
F_DAT = flash memory data
Compare flash memory data to valid data (word compare). If not equal, check for Write Error flag. If Flag not set, compare High and Low Bytes in the Subroutine.
Check buffer of I/O port for more data to write
Reset device for read operation
Turn off V_{PP}

290412-6

Figure 6. Write Algorithm for Word-Wide Mode



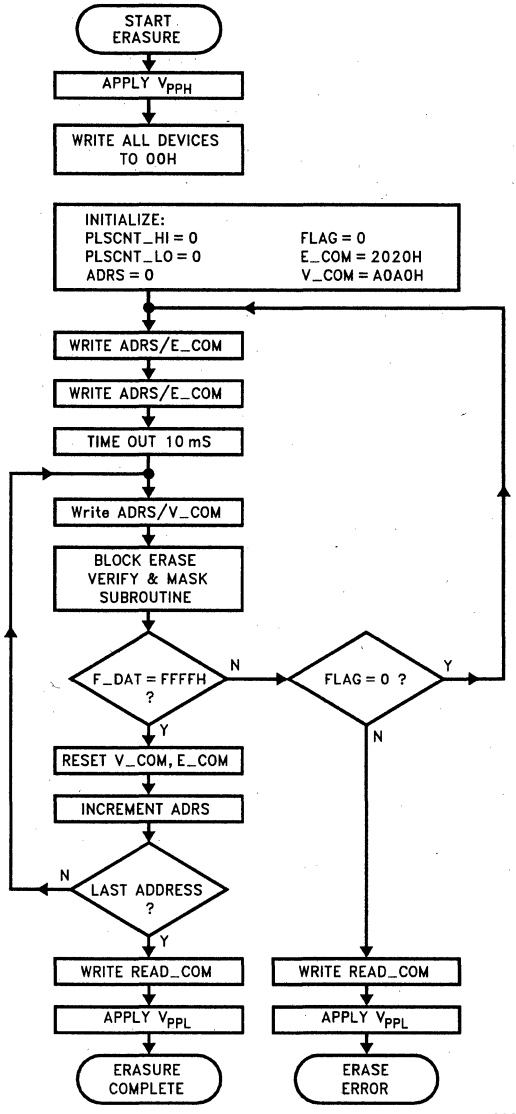
290412-7

Comments
To look at the LO Byte, Mask* the HI Byte with 00
If the LO Byte verifies, mask the LO Byte commands with the reset command (FFH)
If the LO Byte does not verify, then increment its pulse counter and check for max count FLAG = 1 denotes a LO Byte error
Repeat the sequence for the HI Byte
FLAG = 2 denotes a HI Byte error FLAG = 3 denotes both a HI and LO Byte errors. Flag = 0 denotes no max count errors; continue with algorithm.

*Masking can easily and efficiently be done in assembly languages. Simply load word registers with the incoming data (F__DAT), the program commands and the verify commands. Then manipulate the HI or LO register contents.

Figure 7. Write Verify and Mask Subroutine for Word-Wide Mode

6

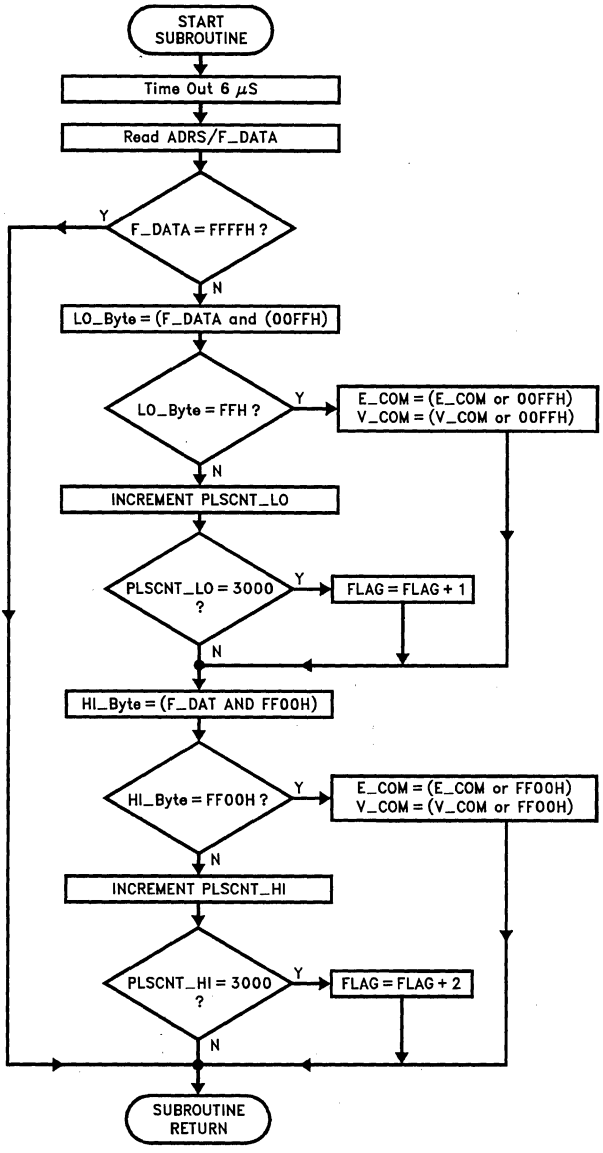


290412-8

Comments
Wait for V_{PP} to stabilize.
Use Write operation algorithm in x8 or x16 configuration
Initialize Variables: PLSCNT_HI = HI Byte Pulse Counter PLSCNT_LO = LO Byte Pulse Counter FLAG = Erasure error flag ADRS = Address E_COM = Erase Command V_COM = Verify Command
Erase Set-up Command
Start Erasing
Duration of Erase Operation
Erase Verify Command stops erasure
See Block Erase Verify & Mask Subroutine
When both devices at ADRS are erased, F_DATA = FFFFH. If not equal, increment the pulse counter and check for last pulse
Reset commands default to (E_COM = 2020H) (V_COM = A0A0H) before verifying next ADRS
Reset device for read operation
Turn off V_{PP}

NOTE:
X16 Addressing uses A_1–A_{20} only. A_0 = 0 throughout word-wide operation.

Figure 8. Erase Algorithm for Word-Wide Mode



290412-9

Comments
This subroutine reads the data word (F__DATA). It then masks the HI or LO Byte of the Erase and Verify commands from executing during the next operation.
If both HI and LO Bytes verify, then return.
Mask* the HI Byte with 00H.
If the LO Byte verifies erasure, then mask* the next erase and verify commands with FFH (RESET).
If the LO Byte does not verify, then increment its pulse counter and check for max count. FLAG = 1 denotes a LO Byte error.
Repeat the sequence for the HI Byte.
Flag = 2 denotes a HI Byte error. Flag = 3 denotes both a HI and LO Byte errors. FLAG = 0 denotes no max count errors; continue with algorithm.

*Masking can easily and efficiently be done in assembly languages. Simply load word registers with the incoming data (F__DAT), the program commands and the verify commands. Then manipulate the HI or LO register contents.

Figure 9. Erase Verify and Mask Subroutine for Word-Wide Mode

6

SYSTEM DESIGN CONSIDERATIONS

Three-Line Control

Three-line control provides for:

a. the lowest possible power dissipation and.

b. complete assurance that output bus contention will not occur.

To efficiently use these three control inputs, an address-decoder output should drive $CE_{1,2}$#, while the system's Read signal controls the card OE# signal, and other parallel zones. This, coupled with the internal zone decoder, assures that only enabled memory zones have active outputs, while deselected zones maintain the low power standby condition.

Power-Supply Decoupling

Flash memory power-switching characteristics require careful device decoupling. System designers are interested in three supply current issues—standby, active and transient current peaks, produced by falling and rising edges of $CE_{1/2}$#. The capacitive and inductive loads on the card and internal flash memory zones determine the magnitudes of these peaks.

Three-line control and proper decoupling capacitor selection will suppress transient voltage peaks. The iMC002FLKA features on-card ceramic decoupling capacitors connected between V_{CC} and V_{SS}, and between V_{PP1}/V_{PP2} and V_{SS}.

The card connector should also have a 4.7 μF electrolytic capacitor between V_{CC} and V_{SS}, as well as between V_{PP1}/V_{PP2} and V_{SS}. The bulk capacitors will overcome voltage slumps caused by printed-circuit-board trace inductance, and will supply charge to the smaller capacitors as needed.

Power Up/Down Protection

The PCMCIA/JEIDA socket is specified, via unique Pin lengths, to properly sequence the power supplies to the flash memory card. This assures that hot insertion and removal will not result in card damage or data loss.

Each zone in the iMC002FLKA is designed to offer protection against accidental erasure or writing, caused by spurious system-level signals that may exist during power transitions. The card will power-up into the read state.

A system designer must guard against active writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and $CE_{1,2}$# must be low for a command write, driving either to V_{IH} will inhibit writes. With its control register architecture, alteration of zone contents only occurs after successful completion of the two-step command sequences.

While these precautions are sufficient for most applications, it is recommended that V_{CC} reach its steady state value before raising $V_{PP1/2}$ above V_{CC} + 2.0V. In addition, upon powering-down, $V_{PP1/2}$ should be below V_{CC} + 2.0V, before lowering V_{CC}.

Absolute Maximum Ratings*

Operating Temperature
During Read .....0°C to + 60°C(1)
During Erase/Write .....0°C to + 60°C

Temperature Under Bias .....−10°C to +70°C

Storage Temperature.....−30°C to +70°C

Voltage on Any Pin with
Respect to Ground .....−2.0V to +7.0V(2)

V_{PP1}/V_{PP2} Supply Voltage with
Respect to Ground
During Erase/Write.....−2.0V to +14.0V(2, 3)

V_{CC} Supply Voltage with
Respect to Ground .....−2.0V to +7.0V(2)

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for commercial product defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to −2.0V for periods less than 20 ns. Maximum DC voltage on output pins is V_{CC} + 0.5V, which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.
3. Maximum DC input voltage on V_{PP1}/V_{PP2} may overshoot to +14.0V for periods less than 20 ns.

OPERATING CONDITIONS

Symbol	Parameter	Limits		Unit	Comments
		Min	Max		
T_A	Operating Temperature	0	60	°C	For Read-Only and Read/Write Operations
V_{CC}	V_{CC} Supply Voltage	4.75	5.25	V	
V_{PPH}	Active V_{PP1}, V_{PP2} Supply Voltages	11.40	12.60	V	
V_{PPL}	V_{PP} During Read Only Operations	0.00	6.50	V	

6

DC CHARACTERISTICS—Byte Wide Mode

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1, 4		±1.0	±20	µA	$V_{CC} = V_{CC}$ max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1		±1.0	±20	µA	$V_{CC} = V_{CC}$ max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1		0.4	0.8	mA	$V_{CC} = V_{CC}$ max $CE_1\# = CE_2\# = V_{CC} \pm 0.2V$
				4	7	mA	$V_{CC} = V_{CC}$ max $CE_1\# = CE_2\# = V_{IH}$
I_{CC1}	V_{CC} Active Read Current	1, 2		25	50	mA	$V_{CC} = V_{CC}$ max $CE\# = V_{IL}$ f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Write Current	1, 2		5.0	15.0	mA	Writing in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		10.0	20.0	mA	Erasure in Progress
I_{CC4}	V_{CC} Write Verify Current	1, 2		10.0	20.0	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress

DC CHARACTERISTICS—Byte Wide Mode (Continued)

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{CC5}	V_{CC} Erase Verify Current	1, 2		10.0	20.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±80	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1, 3		0.4	0.8	mA	$V_{PP} > V_{CC}$
					±0.08		$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Write Current	1, 3		8.0	30	mA	$V_{PP} = V_{PPH}$ Write in Progress
I_{PP3}	V_{PP} Erase Current	1, 3		10	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Write Verify Current	1, 3		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 3		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.4		V_{CC} ± 0.3	V	
V_{OL}	Output Low Voltage				0.40	V	I_{OL} = 3.2 mA $V_{CC} = V_{CC}$ min
V_{OH1}	Output High Voltage		3.8			V	I_{OH} = −2.0 mA $V_{CC} = V_{CC}$ min
V_{PPL}	V_{PP} During Read-Only Operations		0.00		6.5	V	Note: Erase/Write are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} During Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. 1 chip active and 7 in standby for byte-wide mode.
3. Assumes 1 V_{PP} is active.
4. Due to 100 kΩ pull up resistors OE#, CE_1#, CE_2# and WE# will exhibit ≤ 55 μA of additional I_{LI} when $V_{IN} = V_{SS}$.

DC CHARACTERISTICS—Word Wide Mode

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1, 4		±1.0	±20	μA	$V_{CC} = V_{CC}$ max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1		±1.0	±20	μA	$V_{CC} = V_{CC}$ max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1		0.4	0.8	mA	$V_{CC} = V_{CC}$ max CE_1# = CE_2# = V_{CC} ±0.2V
				4	7	mA	$V_{CC} = V_{CC}$ max CE_1# = CE_2# = V_{IH}

DC CHARACTERISTICS—Word Wide Mode (Continued)

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{CC1}	V_{CC} Active Read Current	1, 2		40	80	mA	$V_{CC} = V_{CC}$ max CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Write Current	1, 2		7.0	25	mA	Writing in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		15	30	mA	Erasure in Progress
I_{CC4}	V_{CC} Write Verify Current	1, 2		15	30	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		10	30	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±80	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current	1, 3		0.7	1.6	mA	$V_{PP} > V_{CC}$
	or Standby Current				±0.16		$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Write Current	1, 3		16	60	mA	$V_{PP} = V_{PPH}$ Write in Progress
I_{PP3}	V_{PP} Erase Current	1, 3		20	60	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Write Verify Current	1, 3		5.0	12	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 3		5.0	12	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.4		V_{CC} ± 0.3	V	
V_{OL}	Output Low Voltage				0.40	V	I_{OL} = 3.2 mA $V_{CC} = V_{CC}$ min
V_{OH1}	Output High Voltage		3.8			V	I_{OH} = −2.0 mA $V_{CC} = V_{CC}$ min
V_{PPL}	V_{PP} During Read-Only Operations		0.00		6.5	V	Note: Erase/Write are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} During Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. 2 chips active and 6 in standby for word-wide mode.
3. Assumes 2 V_{PPS} are active.
4. Due to 100 kΩ pull up resistors OE#, CE_1#, CE_2# and WE# will exhibit ≤ 55 μA of additional I_{LI} when $V_{IN} = V_{SS}$.

CAPACITANCE $T = 25°C, f = 1.0$ MHz

Symbol	Parameter	Notes	Limits		Unit	Conditions
			Min	Max		
C_{IN1}	Address Capacitance			40	pF	$V_{IN} = 0V$
C_{IN2}	Control Capacitance			40	pF	$V_{IN} = 0V$
C_{OUT}	Output Capacitance			40	pF	$V_{OUT} = 0V$
$C_{I/O}$	I/O Capacitance			40	pF	$V_{I/O} = 0V$

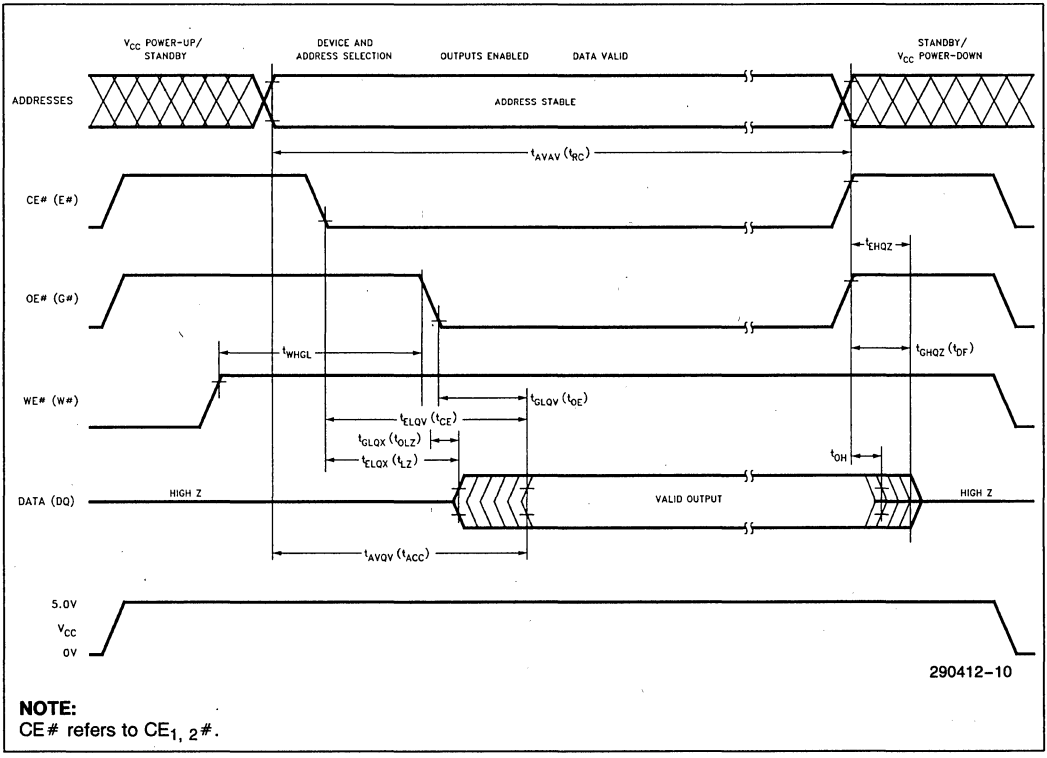
AC TEST CONDITIONS

Input Rise and Fall Times (10% to 90%) ..... 10 ns
Input Pulse Levels ..... V_{OL} and V_{OH1}
Input Timing Reference Level ..... V_{IL} and V_{IH}
Output Timing Reference Level ..... V_{IL} and V_{IH}

AC CHARACTERISTICS—Read-Only Operations

Symbol	Characteristic	Notes	Min	Max	Unit
t_{AVAV}/t_{RC}	Read Cycle Time	2	200		ns
t_{ELQV}/t_{CE}	Chip Enable Access Time	2		200	ns
t_{AVQV}/t_{ACC}	Address Access Time	2		200	ns
t_{GLQV}/t_{OE}	Output Enable Access Time	2		100	ns
t_{ELQX}/t_{LZ}	Chip Enable to Output in Low Z	2	5		ns
t_{EHQZ}	Chip Disable to Output in High Z	2		60	ns
t_{GLQX}/t_{OLZ}	Output Enable to Output in Low Z	2	5		ns
t_{GHQZ}/t_{DF}	Output Disable to Output in High Z	2		60	ns
t_{OH}	Output Hold from Address, CE#, or OE# Change	1, 2	5		ns
t_{WHGL}	Write Recovery Time before Read	2	6		μs

NOTES:
1. Whichever occurs first.
2. Rise/Fall Time ≤ 10 ns.



NOTE:
CE# refers to $CE_{1,2}$#.

Figure 10. AC Waveforms for Read Operations

6

AC CHARACTERISTICS—For Write/Erase Operations

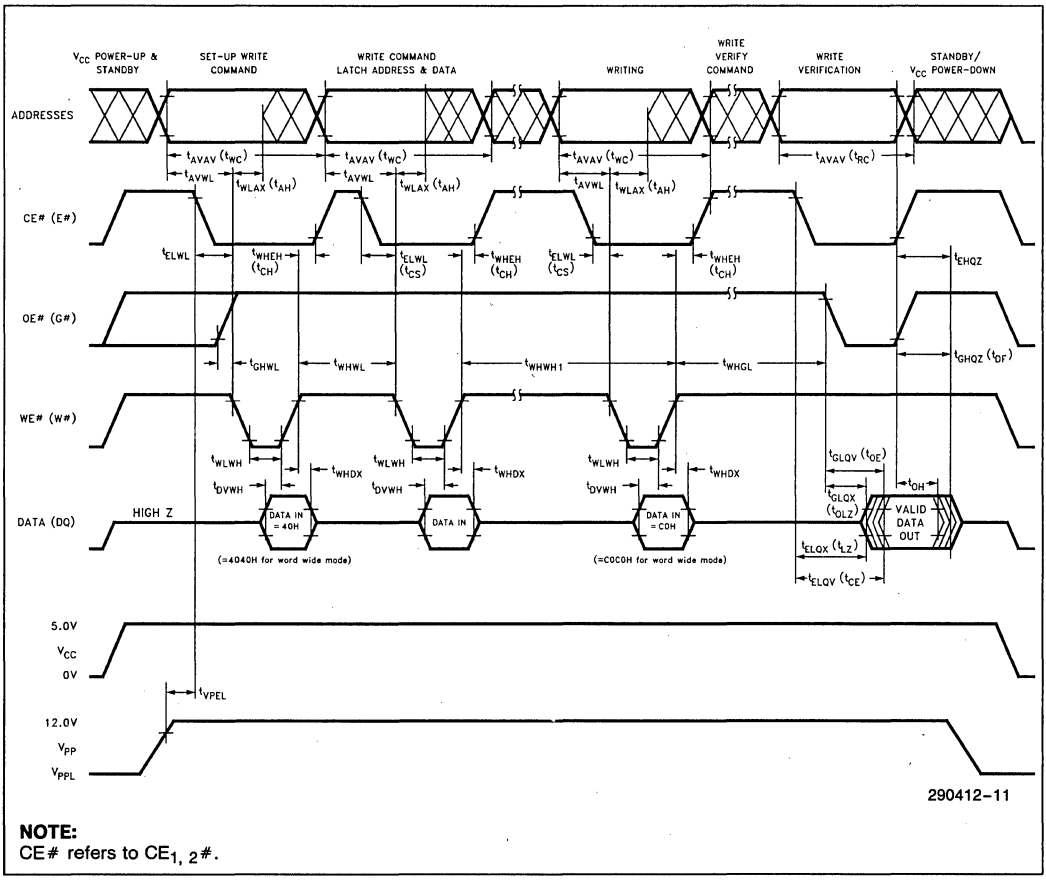
Symbol	Characteristic	Notes	Min	Max	Unit
t_{AVAV}/t_{WC}	Write Cycle Time	1, 2	200		ns
t_{AVWL}/t_{AS}	Address Set-up Time	1, 2	0		ns
t_{WLAX}/t_{AH}	Address Hold Time	1, 2	100		ns
t_{DVWH}/t_{DS}	Data Set-up Time	1, 2	80		ns
t_{WHDX}/t_{DH}	Data Hold Time	1, 2	30		ns
t_{WHGL}	Write Recovery Time before Read	1, 2	6		μs
t_{GHWL}	Read Recovery Time before Write	1, 2	0		μs
t_{WLOZ}	Output High-Z from Write Enable	1, 2	5		ns
t_{WHOZ}	Output Low-Z from Write Enable	1, 2		60	ns
t_{ELWL}/t_{CS}	Chip Enable Set-up Time before Write	1, 2	40		ns
t_{WHEH}/t_{CH}	Chip Enable Hold Time	1, 2	0		ns
t_{WLWH}/t_{WP}	Write Pulse Width	1, 2	100		ns
t_{WHWL}/t_{WPH}	Write Pulse Width High	1, 2	20		ns
t_{WHWH1}	Duration of Write Operation	1, 2, 3	10		μs
t_{WHWH2}	Duration of Erase Operation	1, 2, 3	9.5		ms
t_{VPEL}	V_{PP} Set-up Time to Chip Enable Low	1, 2	100		ns

NOTES:
1. Read timing parameters during read/write operations are the same as during read-only operations. Refer to A.C. Characteristics for Read-Only Operations.
2. Rise/Fall time ≤ 10 ns.
3. The integrated stop timer terminates the write/erase operations, thereby eliminating the need for a maximum specifiction.

ERASE/WRITE PERFORMANCE

Parameter	Notes	Min	Typ	Max	Unit
Zone Erase Time	1, 3, 4		2.0	30	sec
Zone Write Time	1, 2, 4		4.0	25	sec
MTBF	5		10^6		Hrs

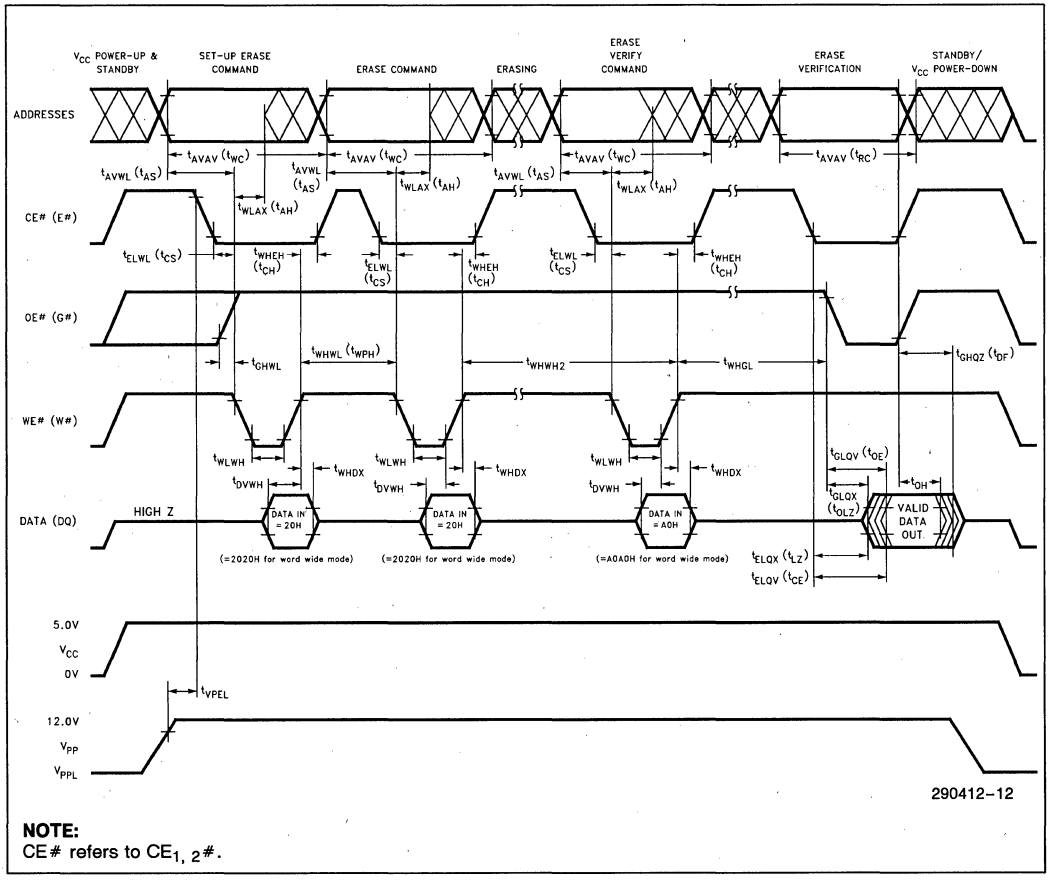
NOTES:
1. 25°C, 12.0V V_{PP}.
2. Minimum byte writing time excluding system overhead is 16 μs (10 μs program + 6 μs write recovery), while maximum is 400 μs/byte (16 μs x 25 loops allowed by algorithm). Max chip write time is specified lower than the worst case allowed by the write algorithm since most bytes write significantly faster than the worst case byte.
3. Excludes 00H writing Prior to Erasure.
4. One zone equals 256 Kbytes.
5. MTBF – Mean Time between Failure, 50% failure point for disk drives.



NOTE:
CE# refers to $CE_{1,2}$#.

Figure 11. AC Waveforms for Write Operations

6



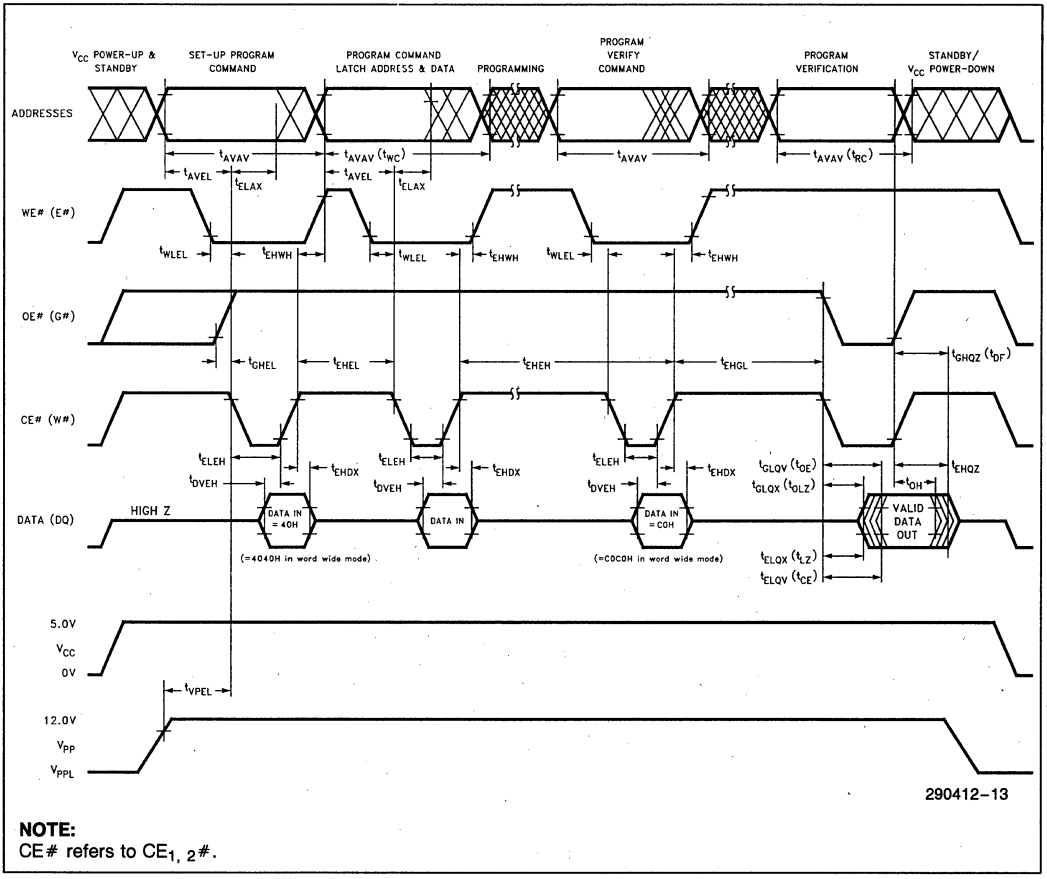
NOTE:
CE# refers to $CE_{1,2}$#.

Figure 12. AC Waveforms for Erase Operations

ALTERNATIVE CE#-CONTROLLED WRITES

Symbol	Characteristic	Notes	Min	Max	Unit
t_{AVAV}	Write Cycle Time		200		ns
t_{AVEL}	Address Set-up Time		0		ns
t_{ELAX}	Address Hold Time		100		ns
t_{DVEH}	Data Set-up Time		80		ns
t_{EHDX}	Data Hold Time		30		ns
t_{EHGL}	Write Recovery Time before Read		6		μs
t_{GHEL}	Read Recovery Time before Write		0		μs
t_{WLEL}	Write Enable Set-Up Time before Chip-Enable		0		ns
t_{EHWH}	Write Enable Hold Time		0		ns
t_{ELEH}	Write Pulse Width	1	100		ns
t_{EHEL}	Write Pulse Width High		20		ns
t_{PEL}	V_{PP} Set-up Time to Chip Enable Low		100		ns

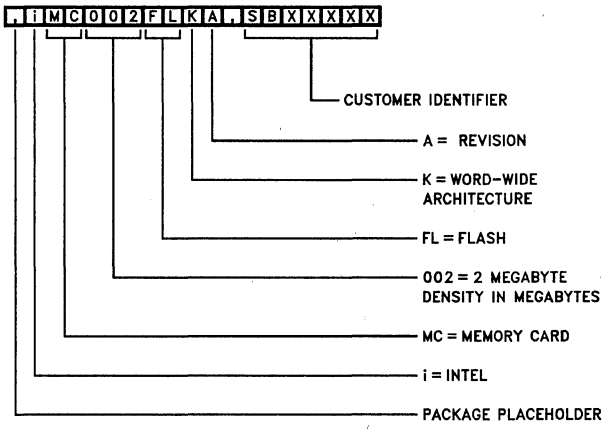
NOTES:
1. Chip Enable Controlled Writes: Write operations are driven by the valid combination of Chip Enable and Write Enable. In systems where Chip Enable defines the write pulse width (with a longer Write Enable timing waveform) all set-up, hold and inactive Write Enable times should be measured relative to the Chip Enable waveform.



NOTE:
CE# refers to $CE_{1,2}$#.

Figure 13. Alternate AC Waveforms for Write Operations

ORDERING INFORMATION



ADDITIONAL INFORMATION	ORDER NUMBER
ER-20, "ETOX™ II Flash Memory Technology"	294005
RR-60, "ETOX™ II Flash Memory Reliability Data Summary"	293002
AP-343, "Solutions for High Density Applications using Flash Memory"	292079

REVISION HISTORY

Number	Description
-002	— Removed Preliminary — Removed ExCA Compliance Section — Clarified need for Valid Address during commands — Corrected $V_{PP} = V_{PPH}$ in Erase Algorithm — Increased I_{CC2}–I_{CC5} D.C. current specs for both Byte Wide and Word Wide modes — Revised and Updated Application Section discussion — Changed order number
-003	— Change Signal with Bar to # — Changed T/C Values

6

iMC001FLKA
1-MBYTE FLASH MEMORY CARD

- **Inherent Nonvolatility (Zero Retention Power)**
  - **— No Batteries Required for Back-Up**
- **High-Performance Read**
  - **— 200 ns Maximum Access Time**
- **CMOS Low Power Consumption**
  - **— 25 mA Typical Active Current (X8)**
  - **— 400 μA Typical Standby Current**
- **Flash Electrical Zone-Erase**
  - **— 1 Second Typical per 128 Kbyte Zone**
  - **— Multiple Zone Erase > 128 KB/s**
- **Random Writes to Erased Zones**
  - **— 10 μs Typical Byte Write**
- **Write Protect Switch to Prevent Accidental Data Loss**
- **Command Register Architecture for Microprocessor/Microcontroller Compatible Write Interface**
- **ETOX™ II Flash Memory Technology**
  - **— 5V Read, 12V Erase/Write**
  - **— High-Volume Manufacturing Experience**
- **PCMCIA/JEIDA 68-Pin Standard**
  - **— Byte- or Word-Wide Selectable**
- **Independent Software & Hardware Vendor Support**
  - **— Integrated System Solution Using Flash Filing Systems**

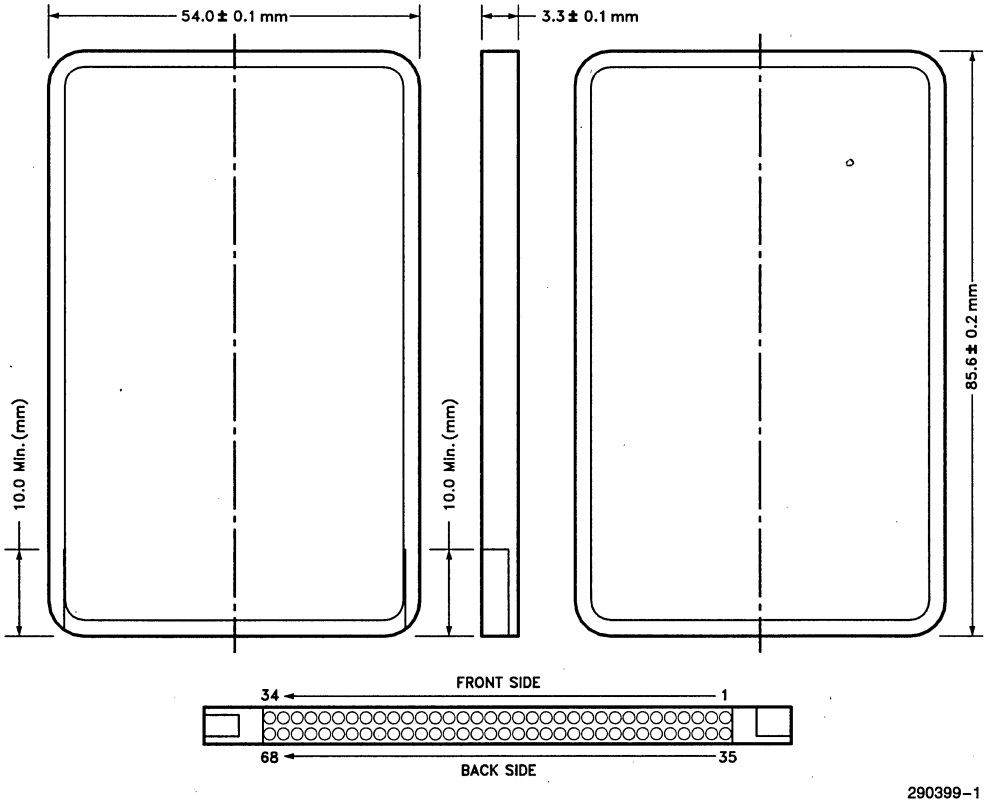
Intel's iMC001FLKA Flash Memory Card is the removable solution for storing and transporting important user data and application code. The combination of rewritability and nonvolatility make the Intel Flash Memory Card ideal for data acquisition and updatable firmware applications. Designing with Intel's Flash Memory Card enables OEM system manufacturers to produce portable and dedicated function systems that are higher performance, more rugged, and consume less power.

The iMC001FLKA conforms to the PCMCIA 1.0 international standard, providing compatibility at the hardware and data interchange level. OEMs may opt to write the Card Information Structure (CIS) at the memory card's address 00000H with a format utility. This information provides data interchange functional compatibility. The 200 ns access time allows for "execute-in-place" capability, for popular low-power microprocessors. Intel's 1-Mbyte Flash Memory Card operates in a byte-wide and word-wide configuration providing performance/power options for different systems.

Intel's Flash Memory card employs Intel's ETOX™ II Flash Memories. Filing systems, such as Microsoft's* Flash File System (FFS), facilitate data file storage and card erasure using a purely nonvolatile medium in the DOS environment. Flash filing systems, coupled with the Intel Flash Memory Card, effectively create an all-silicon nonvolatile read/write random access memory system that is more reliable and higher performance than disk-based memory systems.

*Microsoft is a trademark of Microsoft Corp.

March 1994
Order Number: 290399-004



1	GND	18	V_{PP1}	35	GND	52	V_{PP2}
2	D_3	19	A_{16}	36	CD_1#	53	NC
3	D_4	20	A_{15}	37	D_{11}	54	NC
4	D_5	21	A_{12}	38	D_{12}	55	NC
5	D_6	22	A_7	39	D_{13}	56	NC
6	D_7	23	A_6	40	D_{14}	57	NC
7	CE_1#	24	A_5	41	D_{15}	58	NC
8	A_{10}	25	A_4	42	CE_2#	59	NC
9	OE#	26	A_3	43	NC	60	NC
10	A_{11}	27	A_2	44	NC	61	REG#[1]
11	A_9	28	A_1	45	NC	62	BVD_2#[2]
12	A_8	29	A_0	46	A_{17}	63	BVD_1#[2]
13	A_{13}	30	D_0	47	A_{18}	64	D_8
14	A_{14}	31	D_1	48	A_{19}	65	D_9
15	WE#	32	D_2	49	NC	66	D_{10}
16	NC	33	WP	50	NC	67	CD_2#
17	V_{CC}	34	GND	51	V_{CC}	68	GND

NOTES:
1. REG# = register memory select = No Connect (NC), unused. When REG# is brought low, PCMCIA/JEIDA standard card information structure data is expected. This is accomplished by formatting the card with this data.
2. BVD# = battery detect voltage = Pulled high through pull-up resistor.

Figure 1. iMC001FLKA Pin Configuration

Table 1. Pin Description

Symbol	Type	Name and Function
A_0–A_{19}	I	**ADDRESS INPUTS** for memory locations. Addresses are internally latched during a write cycle.
D_0–D_{15}	I/O	**DATA INPUT/OUTPUT:** Inputs data during memory write cycles; outputs data during memory read cycles. The data pins are active high and float to tri-state OFF when the card is deselected or the outputs are disabled. Data is internally latched during a write cycle.
CE_1#, CE_2#	I	**CARD ENABLE:** Activates the card's high and low byte control logic, input buffers, zone decoders, and associated memory devices. CE# is active low; CE# high deselects the memory card and reduces power consumption to standby levels.
OE#	I	**OUTPUT ENABLE:** Gates the cards output through the data buffers during a read cycle. OE# is active low.
WE#	I	**WRITE ENABLE** controls writes to the control register and the array. Write enable is active low. Addresses are latched on the falling edge and data is latched on the rising edge of the WE# pulse. **NOTE:** With $V_{PP} \leq 6.5V$, memory contents cannot be altered.
V_{PP1}, V_{PP2}		**ERASE/WRITE POWER SUPPLY** for writing the command register, erasing the entire array, or writing bytes in the array.
V_{CC}		**DEVICE POWER SUPPLY** (5V ±5%).
GND		**GROUND**
CD_1#, $CD\#_2$	O	**CARD DETECT:** The card is detected when CD_1# and $CD\#_2$ = ground.
WP	O	**WRITE PROTECT:** All write operations are disabled with WP = active high.
NC		**NO INTERNAL CONNECTION** to device. Pin may be driven or left floating.
BVD_1#, BVD_2#	O	**BATTERY VOLTAGE DETECT:** Not Required.

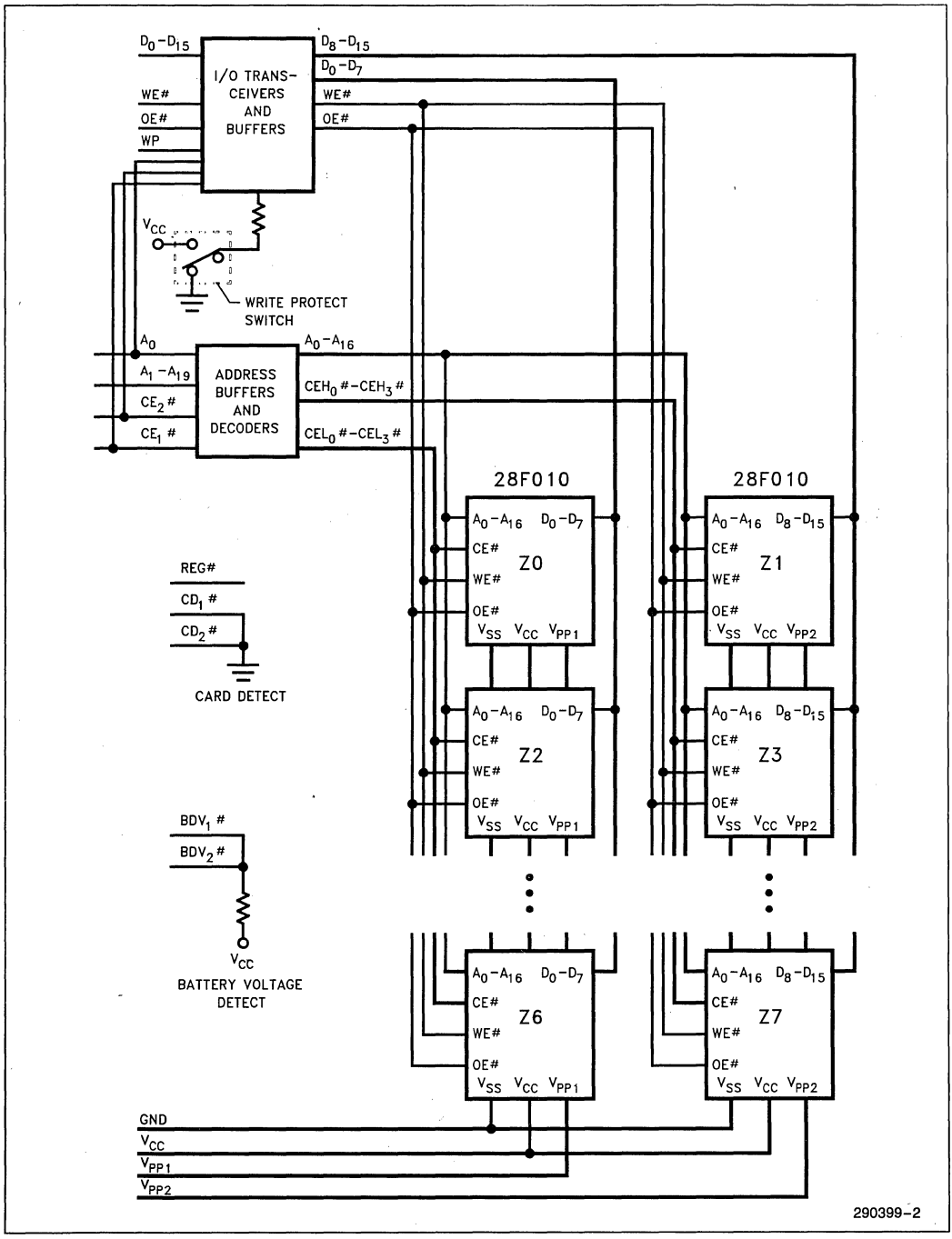


Figure 2. iMC001FLKA Block Diagram

6

APPLICATIONS

The iMC001FLKA Flash Memory Card allows for the storage of data and application programs on a purely solid-state removable medium. System resident flash filing systems, such as Microsoft's Flash File System, allow Intel's ETOX II highly reliable Flash Memory Card to effectively function as a physical disk drive.

User application software stored on the flash memory card substantially reduces the slow disk-DRAM download process. Replacing the disk results in a dramatic enhancement of read performance and substantial reduction of active power consumption, size, and weight—considerations particularly important in portables and dedicated systems. The iMC001FLKA's high performance read access time and command register microprocessor write interface allows for use of the flash memory system in an "execute-in-place" architecture. This configuration eliminates the need for the redundancy associated with DRAM and Disk memory system architectures. ROM based operating systems, such as Microsoft's MS-DOS ROM Version allow for "instant-on" capability. This enables the design of systems that boot, operate, store data files, and execute application code from/to purely nonvolatile memory.

The PCMCIA/JEIDA 68-pin interface enables the end-user to transport data and application code between portables and host systems. Intel Flash Memory Cards provide durable nonvolatile memory storage protecting valuable user code and data.

For systems currently using a static RAM/battery configuration for data acquisition, the iMC001FLKA's inherent nonvolatility eliminates the need for battery backup. The concern for battery failure no longer exists, an important consideration for portable computers and medical instruments, both requiring continuous operation. The iMC001FLKA consumes no power when the system is off. In addition, the iMC001FLKA offers a considerable cost and density advantage over memory cards based on static RAM with battery backup.

The flash memory card's electrical zone-erasure, byte writability, and complete nonvolatility fit well with data accumulation and recording needs. Electrical zone-erasure gives the designer the flexibility to selectively rewrite zones of data while saving other zones for infrequently updated look-up tables, for example.

PRINCIPLES OF OPERATION

Intel's Flash Memory Card combines the functionality of two mainstream memory technologies: the rewritability of RAM and the nonvolatility of EPROM. The flash memory card consists of an array of individual memory devices, each of which defines a physical zone. The iMC001FLKA's memory devices erase as individual blocks, equivalent in size to the 128 Kbyte zone. Multiple zones can be erased simultaneously provided sufficient current for the appropriate number of zones (memory devices). Note, multiple zone erasure requires higher current from both the V_{PP} and V_{CC} power supplies. Erased zones can then be written in bit- or byte-at-a-time fashion and read randomly like RAM. Bit level write capability also supports disk emulation.

In the absence of high voltage on the $V_{PP1/2}$ pins, the iMC001FLKA remains in the read-only mode. Manipulation of the external memory card-control pins yields the standard read, standby, and output disable operations.

The same read, standby, and output disable operations are available when high voltage is applied to the $V_{PP1/2}$ pins. In addition, high voltage on $V_{PP1/2}$

enables erasure and rewriting of the accessed zone(s). All functions associated with altering zone contents—erase, erase verify, write, and write verify—are accessed via the command register.

Commands are written to the internal memory register(s), decoded by zone size, using standard microprocessor write timings. Register contents for a given zone serve as input to that zone's internal state-machine which controls the erase and rewrite circuitry. Write cycles also internally latch addresses and data needed for write and erase operations. With the appropriate command written to the register(s), standard microprocessor read timings output zone data, or output data for erase and write verification.

Byte-Wide or Word-Wide Selection

The flash memory card can be read, erased, and written in a byte-wide or word-wide mode. In the word-wide configuration V_{PP1} and/or $CE_1\#$ control the LO-Byte while V_{PP2} and $CE_2\#$ control the HI-Byte (A_0 = don't care).

Read, Write, and Verify operations are byte- or word-oriented, thus zone independent. Erase Setup and Begin Erase Commands are zone dependent such that commands written to any address within a 128 Kbyte zone boundary initiate the erase operation in that zone (or two 128 Kbyte zones under word-wide operation).

Conventional x8 operation uses $CE_1\#$ active-low, with $CE_2\#$ high, to read or write data through the D_0–D_7 only. "Even bytes" are accessed when A_0 is low, corresponding to the low byte of the complete x16 word. When A_0 is high, the "odd byte" is accessed by transposing the high byte of the complete x16 word onto the D_0–D_7 outputs. This odd byte corresponds to data presented on D_8–D_{15} pins in x16 mode.

Note that two zones logically adjacent in x16 mode are multiplexed through D_0–D_7 in x8 mode and are toggled by the A_0 address. Thus, zone specific erase operations must be kept discrete in x8 mode by addressing even bytes only for one-half of the zone pair, then addressing odd bytes only for the other half.

Card Detection

The flash memory card features two card detect pins ($CD_{1/2}\#$) that allow the host system to determine if the card is properly loaded. Note that the two pins are located at opposite ends of the card. Each CD# output should be read through a port bit. Should only one of the two bits show the card to be present, then the system should instruct the user to re-insert the card squarely into the socket. Card detection can also tell the system whether or not to redirect drives in the case of system booting. $CD_{1/2}\#$ is active low, internally tied to ground.

Write Protection

The flash memory card features three types of write protection. The first type features a mechanical Write Protect Switch that disables the circuitry that control Write Enable to the flash devices. When the switch is activated, the WE# internal to the card is forced high, which disables any writes to the Command Register. The second type of write protection is based on the PCMCIA/JEIDA socket. Unique pin length assignments provide protective power supply sequencing during hot insertion and removal. The third type operates via software control through the Command Register when the card resides in its connector. The Command Register of each zone is only active when $V_{PP1/2}$ is at high voltage. Depending upon the application, the system designer may choose to make $V_{PP1/2}$ power supply switchable—available only when writes are desired. When $V_{PP1/2}$ = V_{PPL}, the contents of the register default to the read command, making the iMC001FLKA a read-only memory card. In this mode, the memory contents cannot be altered.

The system designer may choose to leave $V_{PP1/2}$ = V_{PPH}, making the high voltage supply constantly available. In this case, all Command Register functions are inhibited whenever V_{CC} is below the write lockout voltage, V_{LKO}. (See the section on Power Up/Down Protection.) The iMC001FLKA is designed to accommodate either design practice, and to encourage optimization of the processor-memory card interface.

6

BUS OPERATIONS

Read

The iMC001FLKA has two control functions, both of which must be logically active, to obtain data at the

outputs. Card Enable (CE#) is the power control and should be used for high and/or low zone(s) selection. Output Enable (OE#) is the output control and should be used to gate data from the output pins, independent of accessed zone selection. In the byte-wide configuration, only one CE# is required. The word-wide configuration requires both CE#s active low.

When $V_{PP1/2}$ is high (V_{PPH}), the read operations can be used to access zone data and to access data for write/erase verification. When $V_{PP1/2}$ is low (V_{PPL}), only read accesses to the zone data are allowed.

Output Disable

With Output Enable at a logic-high level (V_{IH}), output from the card is disabled. Output pins are placed in a high-impedance state.

Standby

With one Card Enable at a logic-high level, the standby operation disables one-half of the x16 output's read/write buffer. Further, only the zone corresponding to the selected address within the upper or lower $CE_{1,2}$# bank is active at a time. (**NOTE:** A_0 must be low to select the low half of the x16 word when CE_2# = 1 and CE_1# = 0.) All other zones are deselected, substantially reducing card power consumption. For deselected banks, the outputs are placed in a high-impedance state, independent of the Output Enable signal. If the iMC001FLKA is deselected during erasure, writing, or write/erase verification, the accessed zone draws active current until the operation is terminated.

Intelligent Identifier Command

The manufacturer- and device-codes can be read via the Command Register, for instances where the iMC001FLKA is erased and rewritten in a universal reader/writer. Following a write of 90H to a zone's Command Register, a read from address location 00000H on any zone outputs the manufacturer code (89H). A read from address 0002H outputs the memory device code (B4H).

Write

Zone erasure and rewriting are accomplished via the Command Register, when high voltage is applied to $V_{PP1/2}$. The contents of the register serve as input to that zone's internal state-machine. The state-machine outputs dictate the function of the targeted zone.

The Command Register itself does not occupy an addressable memory location. The register is a latch used to store the command, along with address and data information needed to execute the command.

The Command Register is written by bringing Write Enable to a logic-low level (V_{IL}), while Card Enable(s) is/are low. Addresses are latched on the falling edge of Write Enable, while data is latched on the rising edge of the Write Enable pulse. Standard microprocessor write timings are used.

Refer to AC Write Characteristics and the Erase/Write Waveforms for specific timing parameters.

COMMAND DEFINITIONS

When low voltage is applied to the V_{PP} pins(s), the contents of the zone Command Register(s) default to 00H, enabling read-only operations.

Placing high voltage on the V_{PP} pin(s) enable(s) read/write operations. Zone operations are selected by writing specific data patterns into the Command Register. Tables 3 and 4 define these iMC001FLKA register commands for both byte-wide and word-wide configurations.

All commands written to the Command Register require that the zone address be valid or the incorrect zone will receive the command. Any Command/Data Write or Data Read requires the correct valid address.

Table 2. Bus Operations

Operation	Pins	Notes	[1, 7] V_{PP2}	[1, 7] V_{PP1}	A_0	CE_2#	CE_1#	OE#	WE#	D_8–D_{15}	D_0–D_7
Read-Only	Read (x8)	8	V_{PPL}	V_{PPL}	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Even
	Read (x8)	9	V_{PPL}	V_{PPL}	V_{IH}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Odd
	Read (x8)	10	V_{PPL}	V_{PPL}	X	V_{IL}	V_{IH}	V_{IL}	V_{IH}	Data Out	Tri-state
	Read (x16)	11	V_{PPL}	V_{PPL}	X	V_{IL}	V_{IL}	V_{IL}	V_{IH}	Data Out	Data Out
	Output Disable		V_{PPL}	V_{PPL}	X	X	X	V_{IH}	V_{IH}	Tri-state	Tri-state
	Standby		V_{PPL}	V_{PPL}	X	V_{IH}	V_{IH}	X	X	Tri-state	Tri-state
Read/Write	Read (x8)	3, 8	V_{PPX}	V_{PPH}	V_{IL}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Even
	Read (x8)	3, 9	V_{PPH}	V_{PPX}	V_{IH}	V_{IH}	V_{IL}	V_{IL}	V_{IH}	Tri-state	Data Out-Odd
	Read (x8)	10	V_{PPH}	V_{PPX}	X	V_{IL}	V_{IH}	V_{IL}	V_{IH}	Data Out	Tri-state
	Read (x16)	3, 11	V_{PPH}	V_{PPH}	X	V_{IL}	V_{IL}	V_{IL}	V_{IH}	Data Out	Data Out
	Write (x8)	5, 8	V_{PPX}	V_{PPH}	V_{IL}	V_{IH}	V_{IL}	V_{IH}	V_{IL}	Tri-state	Data In-Even
	Write (x8)	9	V_{PPH}	V_{PPX}	V_{IH}	V_{IH}	V_{IL}	V_{IH}	V_{IL}	Tri-state	Data In-Odd
	Write (x8)	10	V_{PPH}	V_{PPX}	X	V_{IL}	V_{IH}	V_{IH}	V_{IL}	Data In	Tri-state
	Write (x16)	11	V_{PPH}	V_{PPH}	X	V_{IL}	V_{IL}	V_{IH}	V_{IL}	Data In	Data In
	Standby	4	V_{PPH}	V_{PPH}	X	V_{IH}	V_{IH}	X	X	Tri-state	Tri-state
	Output Disable		V_{PPH}	V_{PPH}	X	X	X	V_{IH}	V_{IL}	Tri-state	Tri-state

NOTES:
1. Refer to DC Characteristics. When $V_{PP1/2} = V_{PPL}$ memory contents can be read but not written or erased.
2. Manufacturer and device codes may be accessed via a command register write sequence. Refer to Table 3. All other addresses low.
3. Read operations with $V_{PP1/2} = V_{PPH}$ may access array data or the Intelligent Identifier codes.
4. With $V_{PP1/2}$ at high voltage, the standby current equals $I_{CC} + I_{PP}$ (standby).
5. Refer to Table 3 for valid Data-In during a write operation.
6. X can be V_{IL} or V_{IH}.
7. $V_{PPX} = V_{PPH}$ or V_{PPL}.
8. This x8 operation reads or writes the low byte of the x16 word on DQ_{0-7}, i.e., A_0 low reads "even" byte in x8 mode.
9. This x8 operation reads or writes the high byte of the x16 word on DQ_{0-7} (transposed from DQ_{8-15}), i.e., A_0 high reads "odd" byte in x8 mode.
10. This x8 operation reads or writes the high byte of the x16 on DQ_{8-15}. A_0 is "don't care".
11. A_0 is "don't care", unused in x16 mode. High and low bytes are presented simultaneously.

Table 3. Command Definitions Byte-Wide Mode

Command	Notes	Bus Cycles Req'd	First Bus Cycle			Second Bus Cycle		
			Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory		1	Write	RA	00H			
Read Intelligent Identifier Codes	4	3	Write	IA	90H	Read		
Set-Up Erase/Erase	5	2	Write	ZA	20H	Write	ZA	20H
Erase Verify	5	2	Write	EA	A0H	Read	EA	EVD
Set-Up Write/Write	6	2	Write	WA	40H	Write	WA	WD
Write Verify	6	2	Write	WA	C0H	Read	WA	WVD
Reset	7	2	Write	ZA	FFH	Write	ZA	FFH

Table 4. Command Definitions Word-Wide Mode

Command	Notes	Bus Cycles Req'd	First Bus Cycle			Second Bus Cycle		
			Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory		1	Write	RA	0000H			
Read Intelligent Identifier Codes	4	3	Write	IA	9090H	Read		
Set-Up Erase/Erase	5	2	Write	ZA	2020H	Write	ZA	2020H
Erase Verify	5	2	Write	EA	A0A0H	Read	EA	EVD
Set-Up Write/Write	6	2	Write	WA	4040H	Write	WA	WD
Write Verify	6	2	Write	WA	C0C0H	Read	WA	WVD
Reset	7	2	Write	ZA	FFFFH	Write	ZA	FFFFH

NOTES:
1. Bus operations are defined in Table 2.
2. IA = Identifier address: 00H for manufacturer code, 01H for device code.
   EA = Address of memory location to be read during erase verify.
   RA = Read Address
   WA = Address of memory location to be written.
   ZA = Address of 128 Kbyte zones involved in erase operation.
   Addresses are latched on the falling edge of the Write Enable pulse.
3. ID = Data read from location IA during device indentification. (Mfr = 89H, Device = B4H).
   EVD = Data read from location EA during erase verify.
   WD = Data to be written at location WA. Data is latched on the rising edge of Write Enable.
   WVD = Data read from location WA during write verify. WA is latched on the Write command.
4. Following the Read Intelligent ID command, two read operations access manufacturer and device codes.
5. Figure 5 illustrates the Erase Algorithm.
6. Figure 6 illustrates the Write Algorithm.
7. The second bus cycle must be followed by the desired command register write.
8. The Reset command operation on Zone Basic to Reset entire Card, requires reset Write cycles to each zone.

Read Command

While $V_{PP1/2}$ is high, for erasure and writing, zone memory contents can be accessed via the read command. The read operation is initiated by writing 00H (0000H for the word-wide configuration) into the zone Command Register(s). Microprocessor read cycles retrieve zone data. The accessed zone remains enabled for reads until the Command Register(s) contents are altered.

The default contents of each zone's register(s) upon $V_{PP1/2}$ power-up is 00H (00000H for word-wide). This default value ensures that no spurious alteration of memory card contents occurs during the $V_{PP1/2}$ power transition. Where the $V_{PP1/2}$ supply is left at V_{PPH}, the memory card powers up and remains enabled for reads until the command Register contents of targeted zones are changed. Refer to the AC Read Characteristics and Waveforms for specific timing parameters.

Intelligent Identifier Command

Each zone of the iMC001FLKA contains an Intelligent Identifier to identify memory card device characteristics. The operation is initiated by writing 90H (9090H for word-wide) into the Command Register(s) with zone address. Following the command write, a read cycle from address 00000H retrieves the manufacturer code 89H (8989H for word-wide). A read cycle from address 0002H returns the device code B4H (B4B4H for word-wide). To terminate the operation, it is necessary to write another valid command into the register(s).

Set-Up Erase/Erase Commands

Set-Up Erase stages the targeted zone(s) for electrical erasure of all bytes in the zone. The set-up erase operation is performed by writing 20H to the Command Register (2020H for word-wide) with zone address.

To commence zone-erasure, the erase command (20H or 2020H) must again be written to the register(s) with zone address. The erase operation begins with the rising edge of the Write-Enable pulse and terminates with the rising edge of the next Write-Enable pulse (i.e., Erase-Verify Command).

This two-step sequence of set-up followed by execution ensures that zone memory contents are not accidentally erased. Also, zone-erasure can only occur when high voltage is applied to the $V_{PP1/2}$ pins. In the absence of this high voltage, zone memory contents are protected against erasure. Refer to AC Erase Characteristics and Waveforms for specific timing parameters.

Erase-Verify Command

The erase command erases all of the bytes of the zone in parallel. After each erase operation, all bytes in the zone must be individually verified. In byte-mode operations, zones are segregated by A_0 in odd and even banks; erase and erase verify operations must be done in complete passes of even-bytes-only then odd-bytes-only. See the Erase Algorithm for byte-wide mode. The erase verify operation is initiated by writing A0H (A0A0H for word-wide) into the Command Register(s). The address for the byte(s) to be verified must be supplied as it is latched on the falling edge of the Write Enable pulse. The register write terminates the erase operation with the rising edge of its Write Enable pulse.

The enabled zone applies an internally-generated margin voltage to the addressed byte. Reading FFH from the addressed byte indicates that all bits in the byte are erased. Similarly, reading FFFFH from the addressed word indicates that all bits in the word are erased.

The erase-verify command must be written to the Command Register prior to each byte (word) verification to latch its address. The process continues for each byte (word) in the zone(s) until a byte (word) does not return FFH (FFFFH) data, or the last address is accessed.

In the case where the data read is not FFH (FFFFH), another erase operation is performed. (Refer to Set-Up Erase/Erase.) Verification then resumes from the address of the last-verified byte (word). Once all bytes (words) in the zone(s) have been verified, the erase step is complete. The accessed zone can now be written. At this point, the verify operation is terminated by writing a valid command (e.g., Write Set-Up) to the Command Register. The Erase algorithms for byte-wide and word-wide configurations illustrate how commands and bus operations are combined to perform electrical erasure of the iMC001FLKA. Refer to AC Erase Characteristics and Waveforms for specific timing parameters.

Set-Up Write/Write Commands

Set-Up write is a command-only operation that stages the targeted zone for byte writing. Writing 40H (4040H) into the Command Register(s) performs the set-up operation.

Once the write set-up operation is performed, the next Write Enable pulse causes a transition to an active write operation. Addresses are internally latched on the falling edge of the Write Enable pulse. Data is internally latched on the rising edge of the Write Enable pulse. The rising edge of Write Enable also begins the write operation. The write operation terminates with the next rising edge of Write Enable, which is used to write the verify command. Refer to AC Write Characteristics and Waveforms for specific timing parameters.

Write Verify Command

The iMC001FLKA is written on a byte-by-byte or word-by-word basis. Byte or word writing may occur sequentially or at random. Following each write operation, the byte or word just written must be verified.

The write-verify operation is initiated by writing C0H (C0C0H) into the Command Register(s) with correct address. The register write(s) terminate(s) the write operation with the rising edge of its Write Enable pulse. The write-verify operation stages the accessed zone(s) for verification of the byte or word last written. The zone(s) apply(ies) an internally-generated margin voltage to the byte or word. A microprocessor read cycle outputs the data. A successful comparison between the written byte or word and true data means that the byte or word is successfully written. The write operation then proceeds to the next desired byte or word location. The Write algorithms for byte-wide and word-wide configurations illustrate how commands are combined with bus operations to perform byte and word writes. Refer to AC Write Characteristics and Waveforms for specific timing parameters.

Reset Command

A reset command is provided as a means to safely abort the erase- or write-command sequences. Following either set-up command (erase or write) with two consecutive writes of FFH (FFFFH or word-wide) will safely abort the operation. Zone memory contents will not be altered. A valid command must then be written to place the accessed zone in the desired state.

EXTENDED ERASE/WRITE CYCLING

Intel has designed extended cycling capability into its ETOX II flash memory technology enabling a flash memory card with a MTBF that is many times more reliable than rotating disk technology. Resulting improvements in cycling reliability come without increasing memory cell size or complexity. First, an advanced tunnel oxide increases the charge carrying ability ten-fold. Second, the oxide area per cell subjected to the tunneling electric field minimizes the probability of oxide defects in the region. The lower electric field greatly reduces oxide stress and the probability of failure.

WRITE ALGORITHMS

The write algorithm(s) use write operations of 10 μs duration. Each operation is followed by a byte or word verification to determine when the addressed byte or word has been successfully written. The algorithm(s) allows for up to 25 write operations per byte or word, although most bytes and words verify on the first or second operation. The entire sequence of writing and byte/word verification is performed with V_{PP} at high voltage.

ERASE ALGORITHM

The Erase algorithm(s) yield(s) fast and reliable electrical erasure of memory contents. The algorithm employs a closed-loop flow, similar to the write algorithm, to simultaneously remove charge from all bits in the accessed zone(s).

Erasure begins with a read of memory zone contents. Reading FFH (FFFFH) data from the accessed zone(s) can be immediately followed by writing to the desired zone(s).

For zones being erased and rewritten, uniform and reliable erasure is ensured by first writing all bits in the accessed zone to their charged state (data = 00H byte-wide, 00000H word-wide). This is accomplished, using the write algorithm, in approximately two seconds per zone.

Erase execution then continues with an initial erase operation. Erase veification (data = FFH byte-wide, FFFFH word-wide) begins at address 00000H and continues through the zone to the last address, or until data other than FFH (FFFFH) is encountered. (**Note:** byte-wide erase operation requires separate even- and odd-address passes to handle the individual 128 Kbyte zones.) With each erase operation, an increasing number of bytes or words verify to the erased state. Erase efficiency may be improved by storing the address of the last byte or word verified in a register(s). Following the next erase operation, verification starts at that stored address location. Follow this procedure until all bytes in the zone are erased. Then, re-start the procedure for the next zone or word-wide zone pair. Erasure typically occurs in one second per zone.

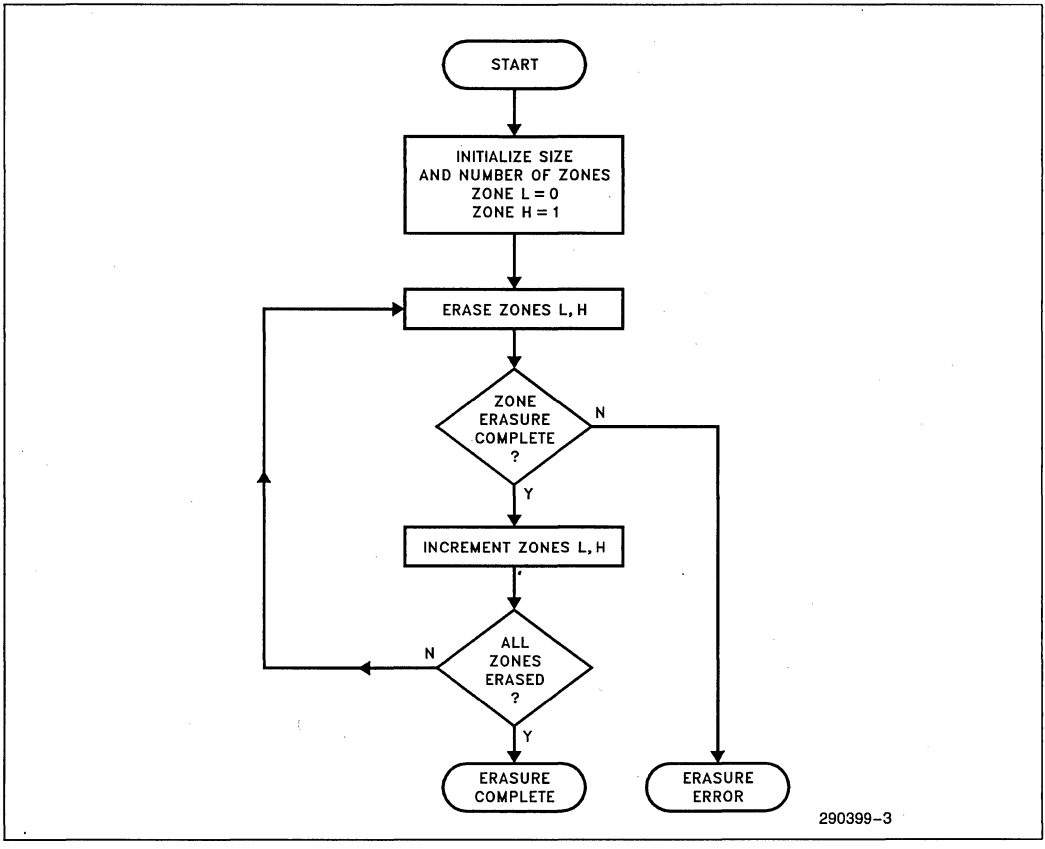
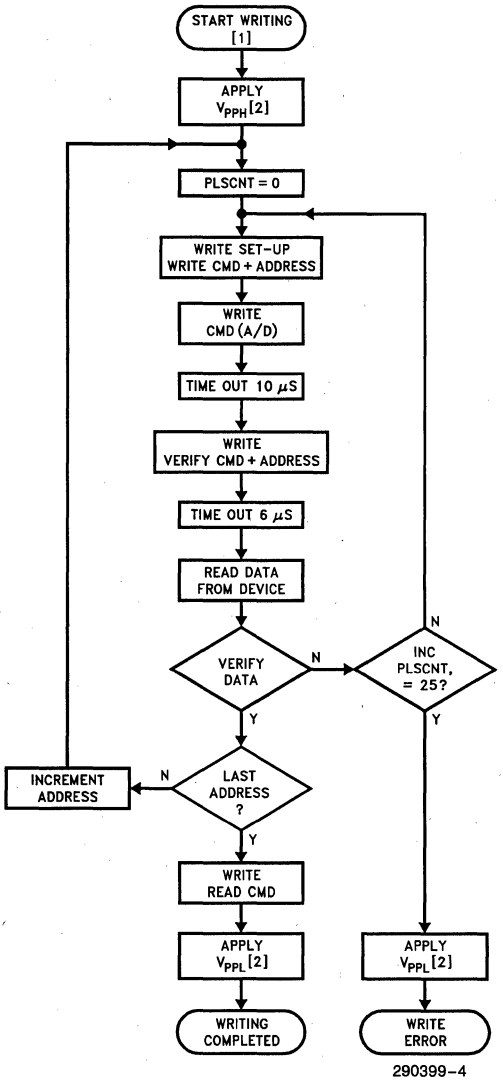


Figure 3. Full Card Erase Flow

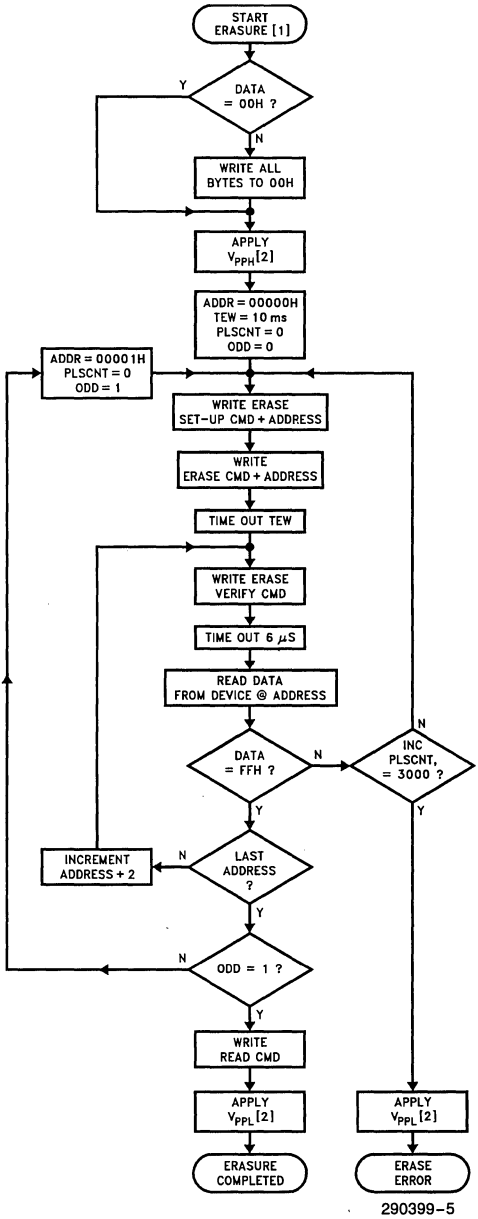
6



Bus Operation	Command	Comments
Standby		Wait for V_{PP} Ramp to V_{PPH} (= 12.0V)(2)
		Initialize Pulse-Count
Write	Set-Up Write	Data = 40H + Valid Address
Write	Write	Valid Address/Data
Standby		Duration of Write Operation (t_{WHWH1})
Write	Write(3) Verify	Data = C0H + Valid Address; Stops(4) Write Operation
Standby		t_{WHGL}
Read		Read Byte to Verify Write Operation at Valid Address
Standby		Compare Data Output to Data Expected
Write	Read	Data = 00H, Resets the Register for Read Operations
Standby		Wait for V_{PP} Ramp to V_{PPL}(2)

NOTES:
1. CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.
2. See D.C. Characteristics for the value of V_{PPH} and V_{PPL}.
3. Write Verify is only performed after a byte write operation. A final read/compare may be performed (optional) after the register is written with the Read command.
4. Refer to principles of operation.

Figure 4. Write Algorithm for Byte-Wide Mode

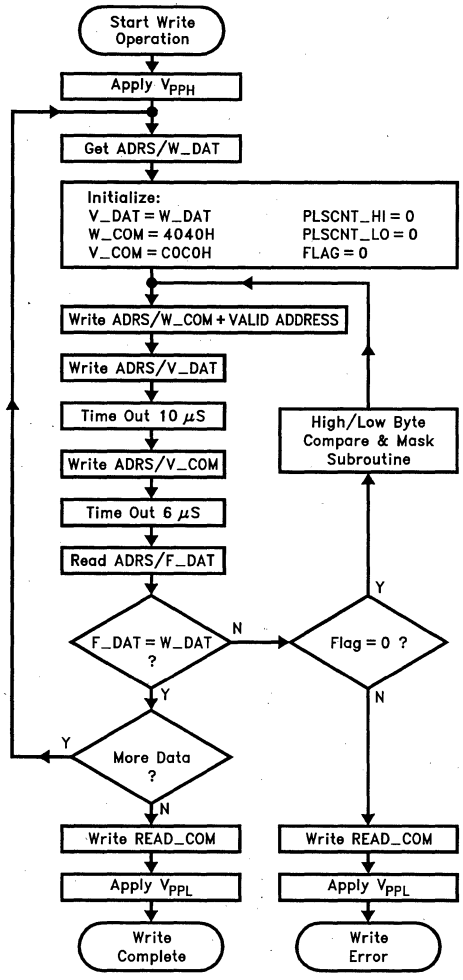


Bus Operation	Command	Comments
Standby		Wait for V_{PP} Ramp to V_{PPH} (= 12.0V)(2)
		Use Write Operation Algorithm
		Initialize Even/Odd Addresses, Erase Pulse Width, and Pulse Count
Write	Set-Up Erase	Data = 20H + Address
Write	Erase	Data = 20H + Address
Standby		Duration of Erase Operation (t_{WHWH2})
Write	Erase Verify(3)	Addr = Byte to Verify; Data = A0H; Stops
Standby		Erase Operation(4) t_{WHGL}
Read		Read Byte to Verify Erasure at Address
Standby		Compare Output to FFH Increment Pulse Count
Write	Read	Data = 00H, Resets the Register for Read Operations
Standby		Wait for V_{PP} Ramp to V_{PPL}(2)

NOTES:
1. CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.
2. See DC Characteristics for the value of V_{PPH} and V_{PPL}.
3. Erase Verify is only performed after chip erasure. A final read/compare may be performed (optional) after the register is written with the Read command.
4. Refer to principles of operation.

Figure 5. Erase Algorithm for Byte-Wide Mode

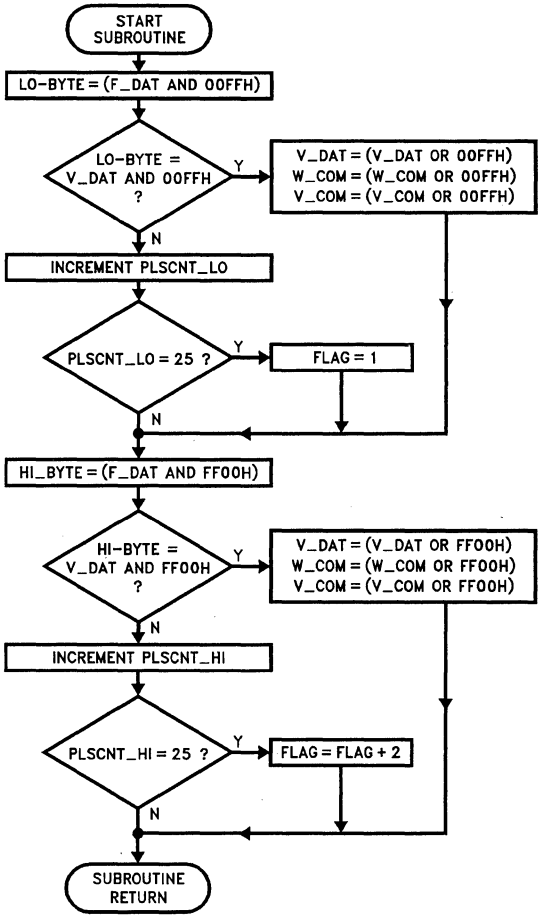
6



Comments
Wait for V_{PP} ramp to V_{PPH}
ADRS = Address to Write W_DAT = Data Word to Write
Initialize Data Word Variables: V_DAT = Valid Data W_COM = Write Command V_COM = Write Verify Command PLSCNT_HI = HI Byte Pulse Counter PLSCNT_LO = LO Byte Pulse Counter FLAG = Write Error Flag
Write Set-Up Command Address needs to be Valid
Write
See Write Verify and Mask Subroutine
Write Verify Command
F_DAT = Flash Memory Data
Compare Flash Memory Data to Valid Data (Word Compare). If Not Equal, Check for Write Error Flag. If Flag Not Set, Compare High and Low Bytes in the Subroutine.
Check Buffer of I/O Port for More Data to Write.
Reset Device for Read Operation.
Turn off V_{PP}.

290399-6

Figure 6. Write Algorithm for Word-Wide Mode



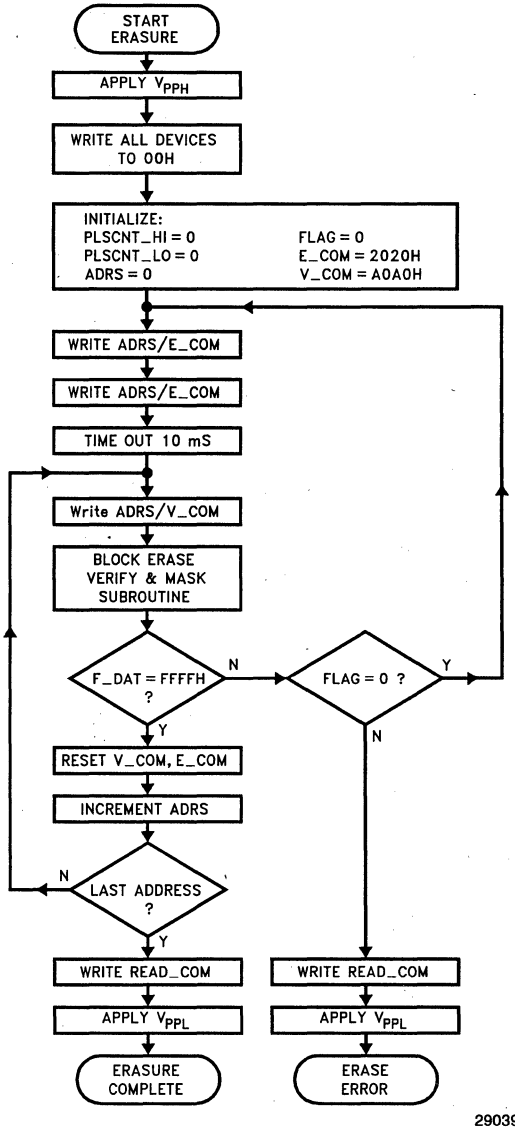
290399-7

Comments
To Look at the LO Byte, Mask* the HI Byte with 00.
If the LO Byte Verifies, Mask the LO Byte Commands with the Reset Command (FFH)
If the LO Byte Does Not Verify, then Increment its Pulse Counter and Check for Max Count. FLAG = 1 Denotes a LO Byte Error.
Repeat the sequence for the HI Byte.
FLAG = 2 Denotes a HI Byte Error. FLAG = 3 Denotes both a HI and LO Byte Errors. FLAG = 0 Denotes No Max Count Errors; Continue with Algorithm.

*Masking can easily and efficiently be done in assembly languages. Simply load word registers with the incoming data (F__DAT), the program commands and the verify commands. Then manipulate the HI or LO register contents.

Figure 7. Write Verify and Mask Subroutine for Word-Wide Mode

6

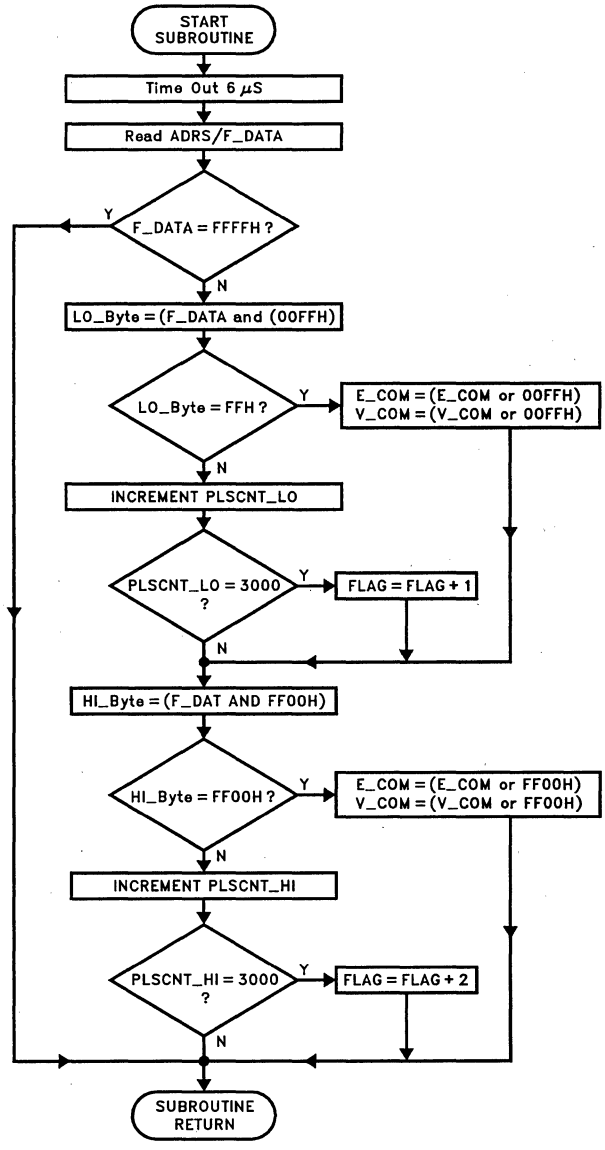


290399-8

Comments
Wait for V_{PP} to stabilize.
Use Write Operation Algorithm in x8 or x16 Configuration.
Initialize Variables: PLSCNT__HI = HI Byte Pulse Counter PLSCNT__LO = LO Byte Pulse Counter FLAG = Erasure Error Flag ADRS = Address E__COM = Erase Command V__COM = Verify Command
Erase Set-Up Command
Start Erasing
Duration of Erase Operation.
Erase Verify Command Stops Erasure.
See Block Erase Verify and Mask Subroutine
When Both Devices at ADRS Are Erased, F__DATA = FFFFH. If Not Equal, Increment the Pulse Counter and Check for Last Pulse.
Reset Commands Default to (E__COM = 2020H) (V__COM = A0A0H) before Verifying Next ADRS.
Reset Device for Read Operation.
Turn Off V_{PP}.

NOTES:
x16 Addressing uses A_1–A_{19} only. A_0 = 0 throughout word-wide operation.

Figure 8. Erase Algorithm for Word-Wide Mode



290399-9

Comments

This subroutine Reads the Data Word (F__DATA). It then masks the HI or LO Byte of the Erase and Verify Commands from Executing during the Next Operation.

If both HI and LO Bytes Verify, then Return.

Mask* the HI Byte with 00H.

If the LO Byte Verifies Erasure, then Mask* the Next Erase and Verify Commands with FFH (RESET).

If the LO Byte Does Not Verify, then Increment its Pulse Counter and Check for Max Count. FLAG = 1 Denotes a LO Byte Error.

Repeat the Sequence for the HI Byte.

FLAG = 2 Denotes a HI Byte Error. FLAG = 3 Denotes both HI and LO Byte Errors. FLAG = 0 Denotes no Max Count Errors; Continue with Algorithm.

*Masking can easily and efficiently be done in assembly languages. Simply load word registers with the incoming data (F__DAT), the program commands and the verify commands. Then manipulate the HI or LO register contents.

Figure 9. Erase Verify and Mask Subroutine for Word-Wide Mode

6

SYSTEM DESIGN CONSIDERATIONS

Three-Line Control

Three-line control provides for:

a. the lowest possible power dissipation and,

b. complete assurance that output bus contention will not occur.

To efficiently use these three control inputs, an address-decoder output should drive $CE_{1,2}$#, while the system's Read signal controls the card OE# signal, and other parallel zones. This, coupled with the internal zone decoder, assures that only enabled memory zones have active outputs, while deselected zones maintain the low power standby condition.

Power-Supply Decoupling

Flash memory power-switching characteristics require careful device decoupling. System designers are interested in three supply current issues—standby, active and transient current peaks, produced by falling and rising edges of $CE_{1/2}$#. The capacitive and inductive loads on the card and internal flash memory zones determine the magnitudes of these peaks.

Three-line control and proper decoupling capacitor selection will suppress transient voltage peaks. The iMC001FLKA features on-card ceramic decoupling capacitors connected between V_{CC} and V_{SS}, and between V_{PP1}/V_{PP2} and V_{SS}.

The card connector should also have a 4.7 μF electrolytic capacitor between V_{CC} and V_{SS}, as well as between V_{PP1}/V_{PP2} and V_{SS}. The bulk capacitors will overcome voltage slumps caused by printed-circuit-board trace inductance, and will supply charge to the smaller capacitors as needed.

Power Up/Down Protection

The PCMCIA/JEIDA socket is specified, via unique Pin lengths, to properly sequence the power supplies to the flash memory card. This assures that hot insertion and removal will not result in card damage or data loss.

Each zone in the iMC001FLKA is designed to offer protection against accidental erasure or writing, caused by spurious system-level signals that may exist during power transitions. The card will power-up into the read state.

A system designer must guard against active writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and $CE_{1,2}$# must be low for a command write, driving either to V_{IH} will inhibit writes. With its control register architecture, alteration of zone contents only occurs after successful completion of the two-step command sequences.

While these precautions are sufficient for most applications, it is recommended that V_{CC} reach its steady state value before raising $V_{PP1/2}$ above V_{CC} + 2.0V. In addition, upon powering-down, $V_{PP1/2}$ should be below V_{CC} + 2.0V, before lowering V_{CC}.

ABSOLUTE MAXIMUM RATINGS*

Operating Temperature
During Read ..... 0°C to +60°C(1)
During Erase/Write ..... 0°C to +60°C

Temperature under Bias ..... −10°C to +70°C

Storage Temperature ..... −30°C to +70°C

Voltage on Any Pin with
Respect to Ground ..... −2.0V to +7.0V(2)

V_{PP1}/V_{PP2} Supply Voltage
with Respect to Ground
during Erase/Write ..... −2.0V to +14.0V(2, 3)

V_{CC} Supply Voltage with
Respect to Ground ..... −2.0V to +7.0V(2)

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for commercial product defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to −2.0V for periods less than 20 ns. Maximum DC voltage on output pins is V_{CC} + 0.5V, which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.
3. Maximum DC input voltage on V_{PP1}/V_{PP2} may overshoot to +14.0V for periods less than 20 ns.

OPERATING CONDITIONS

Symbol	Parameter	Limits		Unit	Comments
		Min	Max		
T_A	Operating Temperature	0	60	°C	For Read-Only and Read/Write Operations
V_{CC}	V_{CC} Supply Voltage	4.75	5.25	V	
V_{PPH}	Active V_{PP1}, V_{PP2} Supply Voltages	11.40	12.60	V	
V_{PPL}	V_{PP} during Read Only Operations	0.00	6.50	V	

6

DC CHARACTERISTICS—Byte Wide Mode

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typ	Max		
I_{LI}	Input Leakage Current	1, 4		±1.0	±20	μA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1		±1.0	±20	μA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1		0.4	0.8	mA	$V_{CC} = V_{CC}$ Max CE# = V_{CC} = ±0.2V
				4	7	mA	CE# = V_{IH}, $V_{CC} = V_{CC}$ Max
I_{CC1}	V_{CC} Active Read Current	1, 2		25	50	mA	$V_{CC} = V_{CC}$, Max CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Write Current	1, 2		5.0	15.0	mA	Writing in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		10.0	20.0	mA	Erasure in Progress
I_{CC4}	V_{CC} Write Verify Current	1, 2		10.0	20.0	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		10.0	20.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±80	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1, 3		0.4	0.8	mA	$V_{PP} > V_{CC}$
					±0.08		$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Write Current	1, 3		8.0	30	mA	$V_{PP} = V_{PPH}$ Write in Progress
I_{PP3}	V_{PP} Erase Current	1, 3		10	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Write Verify Current	1, 3		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 3		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.4		V_{CC} + 0.3	V	
V_{OL}	Output Low Voltage				0.40	V	I_{OL} = 3.2 mA $V_{CC} = V_{CC}$ Min
V_{OH1}	Output High Voltage		3.8			V	I_{OH} = −2.0 mA $V_{CC} = V_{CC}$ Min
V_{PPL}	V_{PP} during Read-Only Operations		0.00		6.5	V	**Note:** Erase/Write are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} during Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. 1 chip active and 7 in standby for byte-wide mode.
3. Assumes 1 V_{PP} is active.
4. Due to 100 kΩ pull up resistors, OE#, CE_1#, CE_2#, and WE# will exhibit ≤ 55 μA of additional I_{LI} when $V_{IN} = V_{SS}$.

DC CHARACTERISTICS—Word Wide Mode

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typ	Max		
I_{LI}	Input Leakage Current	1, 4		±1.0	±20	μA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1		±1.0	±20	μA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1		0.4	0.8	mA	$V_{CC} = V_{CC}$ Max, CE# = V_{CC} ±0.2V
				4	7	mA	$V_{CC} = V_{CC}$ Max, CE# = V_{IH}
I_{CC1}	V_{CC} Active Read Current	1, 2		40	80	mA	$V_{CC} = V_{CC}$, Max CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Write Current	1, 2		5.0	25	mA	Writing in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		15.0	30	mA	Erasure in Progress
I_{CC4}	V_{CC} Write Verify Current	1, 2		15.0	30	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		15.0	30	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±80	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1, 3		0.7	1.6	mA	$V_{PP} \geq V_{CC}$
					±0.16		$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Write Current	1, 3		16	60	mA	$V_{PP} = V_{PPH}$ Write in Progress
I_{PP3}	V_{PP} Erase Current	1, 3		20	60	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Write Verify Current	1, 3		5.0	12	mA	$V_{PP} = V_{PPH}$ Write Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 3		5.0	12	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.4		V_{CC} + 0.3	V	
V_{OL}	Output Low Voltage				0.40	V	I_{OL} = 3.2 mA $V_{CC} = V_{CC}$ Min
V_{OH1}	Output High Voltage		3.8			V	I_{OH} = −2.0 mA $V_{CC} = V_{CC}$ Min
V_{PPL}	V_{PP} during Read-Only Operations		0.00		6.5	V	**Note:** Erase/Write are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} during Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. 2 chips active and 6 in standby for word-wide mode.
3. Assumes 2 V_{PPS} are active.
4. Due to 100 kΩ pull up resistors, OE#, CE_1#, CE_2#, and WE# will exhibit ≤ 55 μA of additional I_{LI} when $V_{IN} = V_{SS}$.

CAPACITANCE $T = 25°C, f = 1.0$ MHz

Symbol	Parameter	Notes	Limits		Unit	Conditions
			Min	Max		
C_{IN1}	Address Capacitance			40	pF	$V_{IN} = 0V$
C_{IN2}	Control Capacitance			40	pF	$V_{IN} = 0V$
C_{OUT}	Output Capacitance			40	pF	$V_{OUT} = 0V$
$C_{I/O}$	I/O Capacitance			40	pF	$V_{I/O} = 0V$

AC TEST CONDITIONS

Input Rise and Fall Times (10% to 90%) ..... 10 ns
Input Pulse Levels ..... V_{OL} and V_{OH1}
Input Timing Reference Level ..... V_{IL} and V_{IH}
Output Timing Reference Level ..... V_{IL} and V_{IH}

AC CHARACTERISTICS—Read-Only Operations

Symbol	Parameter	Notes	Min	Max	Unit
t_{AVAV}/t_{RC}	Read Cycle Time	2	200		ns
t_{ELQV}/t_{CE}	Chip Enable Access Time	2		200	ns
t_{AVQV}/t_{ACC}	Address Access Time	2		200	ns
t_{GLQV}/t_{OE}	Output Enable Access Time	2		100	ns
t_{ELQX}/t_{LZ}	Chip Enable to Output in Low Z	2	5		ns
t_{EHQZ}	Chip Disable to Output in High Z	2		60	ns
t_{GLQX}/t_{OLZ}	Output Enable to Output in Low Z	2	5		ns
t_{GHQZ}/t_{DF}	Output Disable to Output in High Z	2		60	ns
t_{OH}	Output Hold from Address, CE#, or OE# Change	1, 2	5		ns
t_{WHGL}	Write Recovery Time before Read	2	5		μs

NOTES:
1. Whichever occurs first.
2. Rise/Fall time ≤ 10 ns.

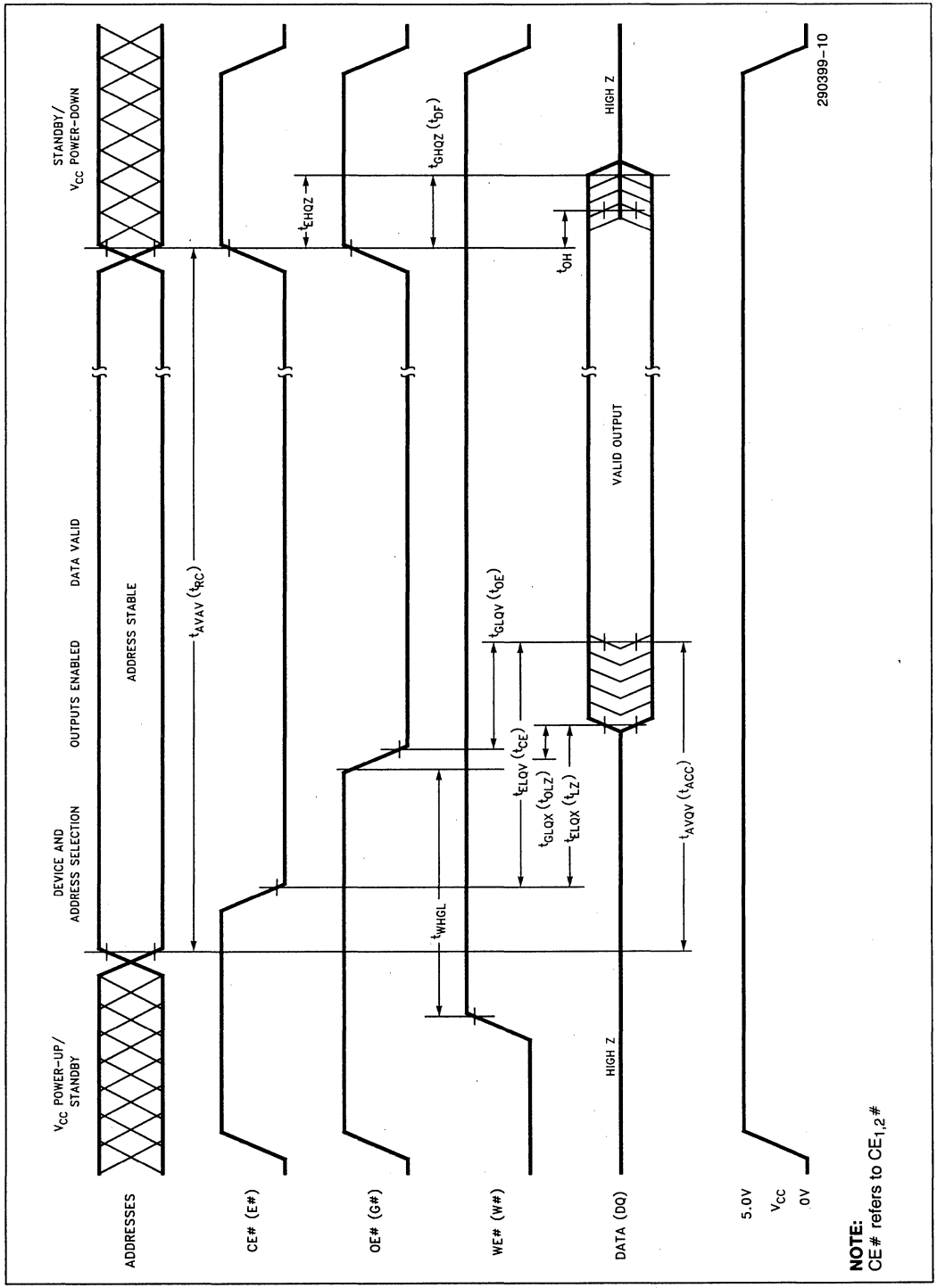


Figure 10. AC Waveforms for Read Operations

AC CHARACTERISTICS—For Write/Erase Operations

Symbol	Parameter	Notes	Min	Max	Unit
t_{AVAV}/t_{WC}	Write Cycle Time	1, 2	200		ns
t_{AVWL}/t_{AS}	Address Set-Up Time	1, 2	0		ns
t_{WLAX}/t_{AH}	Address Hold Time	1, 2	100		ns
t_{DVWH}/t_{DS}	Data Set-Up Time	1, 2	80		ns
t_{WHDX}/t_{DH}	Data Hold Time	1, 2	30		ns
t_{WHGL}	Write Recovery Time before Read	1, 2	6		μs
t_{GHWL}	Read Recovery Time before Write	1, 2	0		μs
t_{WLOZ}	Output High-Z from Write Enable	1, 2	5		ns
t_{WHOX}	Output Low-Z from Write Enable	1, 2		60	ns
t_{ELWL}/t_{CS}	Chip Enable Set-Up Time before Write	1, 2	40		ns
t_{WHEH}/t_{CH}	Chip Enable Hold Time	1, 2	0		ns
t_{WLWH}/t_{WP}	Write Pulse Width	1, 2	100		ns
t_{WHWL}/t_{WPH}	Write Pulse Width High	1, 2	20		ns
t_{WHWH1}	Duration of Write Operation	1, 2, 3	10		μs
t_{WHWH2}	Duration of Erase Operation	1, 2, 3	9.5		ms
t_{VPEL}	V_{PP} Set-Up Time to Chip Enable Low	1, 2	100		ns

NOTES:
1. Read timing parameters during read/write operations are the same as during read-only operations. Refer to AC Characteristics for Read-Only Operations.
2. Rise/Fall time ≤ 10 ns.
3. The integrated stop timer terminates the write/erase operations, thereby eliminating the need for a maximum specification.

ERASE/WRITE PERFORMANCE

Parameter	Notes	Min	Typ	Max	Unit
Zone Erase Time	1, 3, 4		1.0	10	sec
Zone Write Time	1, 2, 4		2.0	12.5	sec
MTBF	5		10^6		Hrs

NOTES:
1. 25°C, 12.0V V_{PP}.
2. Minimum byte writing time excluding system overhead is 16 μs (10 μs program + 6 μs write recovery), while maximum is 400 μs/byte (16 μs × 25 loops allowed by algorithm). Max chip write time is specified lower than the worst case allowed by the write algorithm since most bytes write significantly faster than the worst case byte.
3. Excludes 00H writing Prior to Erasure.
4. One zone equals 128 Kbytes.
5. MTBF = Mean Time Between Failure, 50% failure point for disk drives.

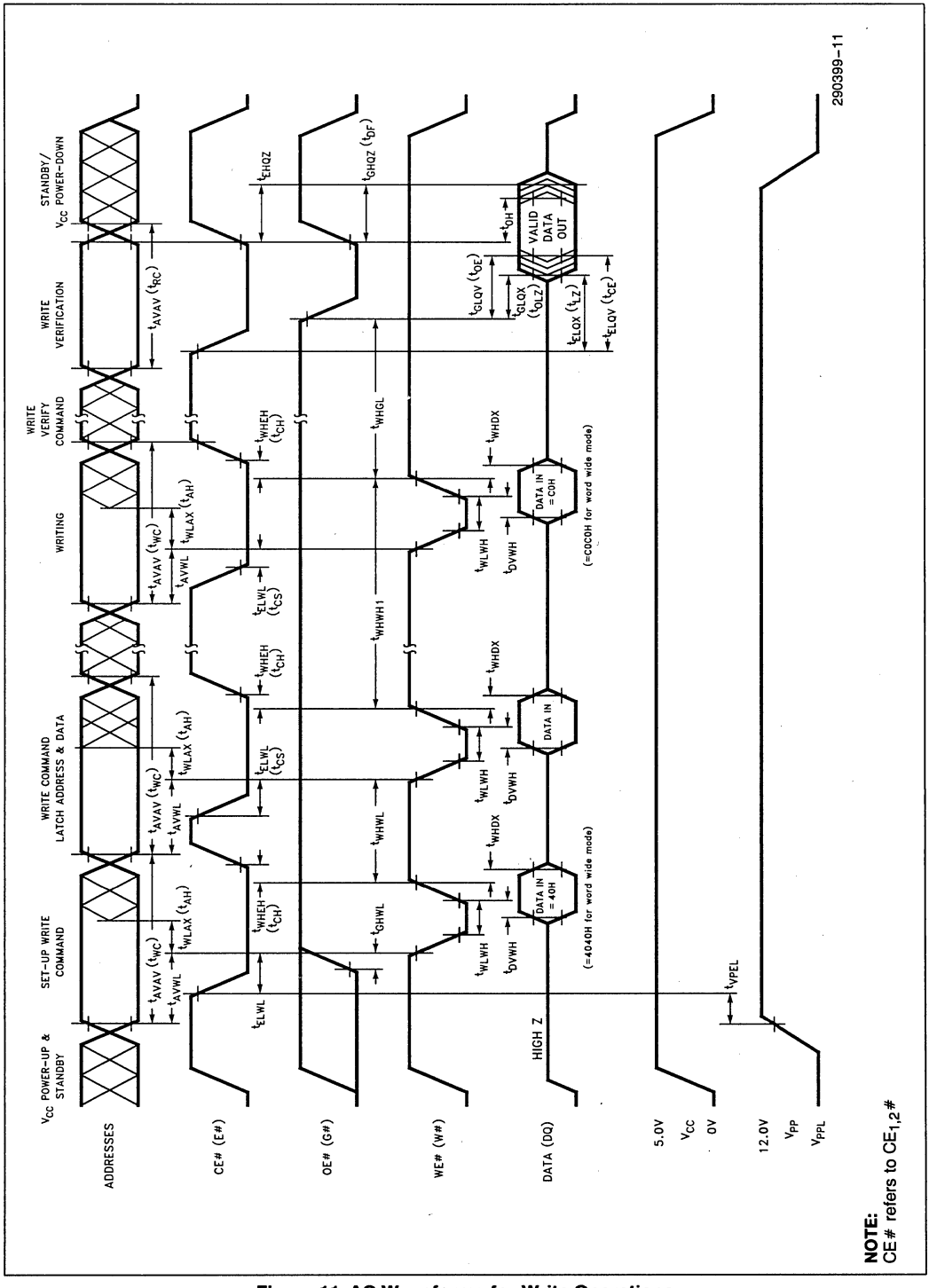


Figure 11. AC Waveforms for Write Operations

6

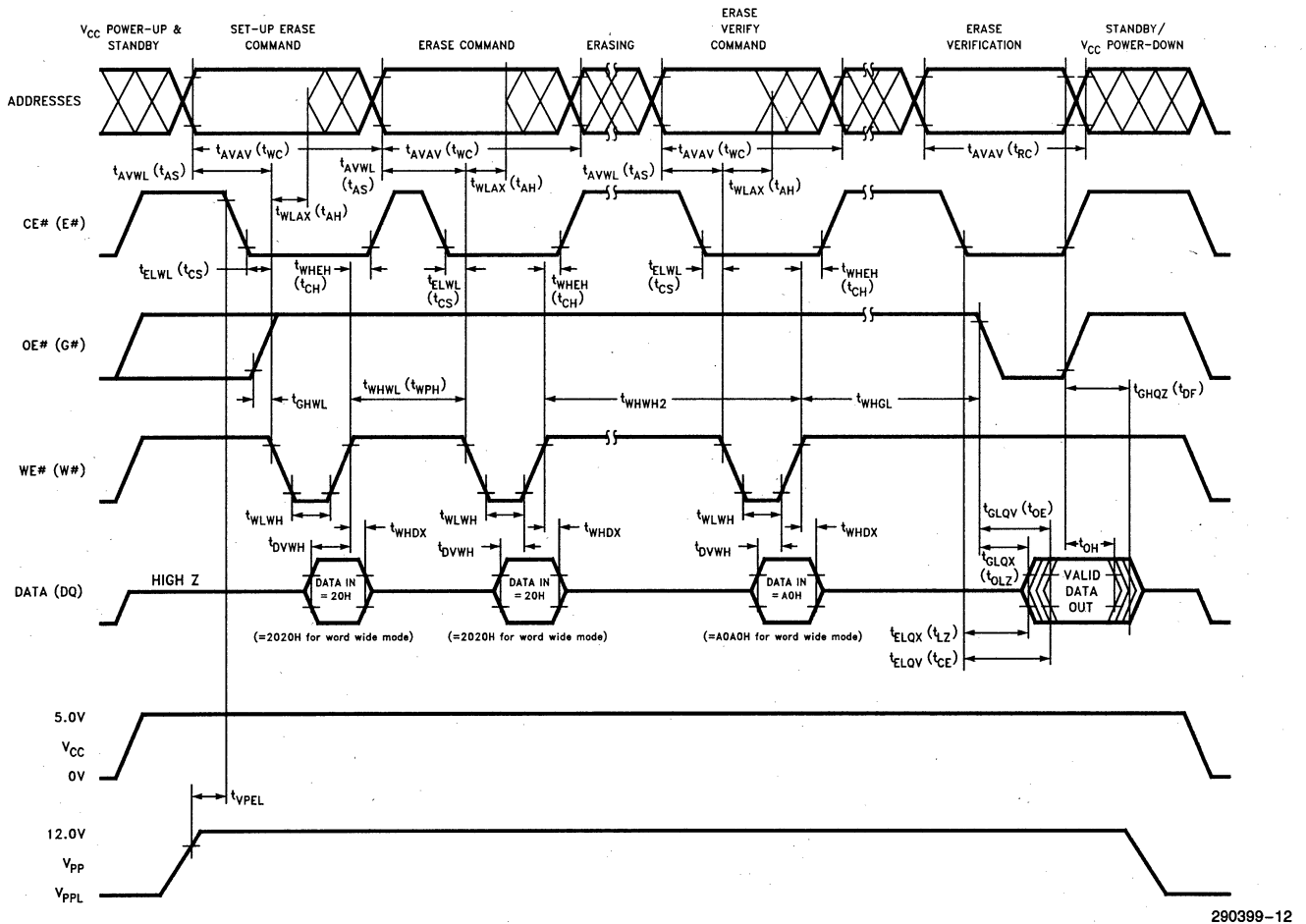


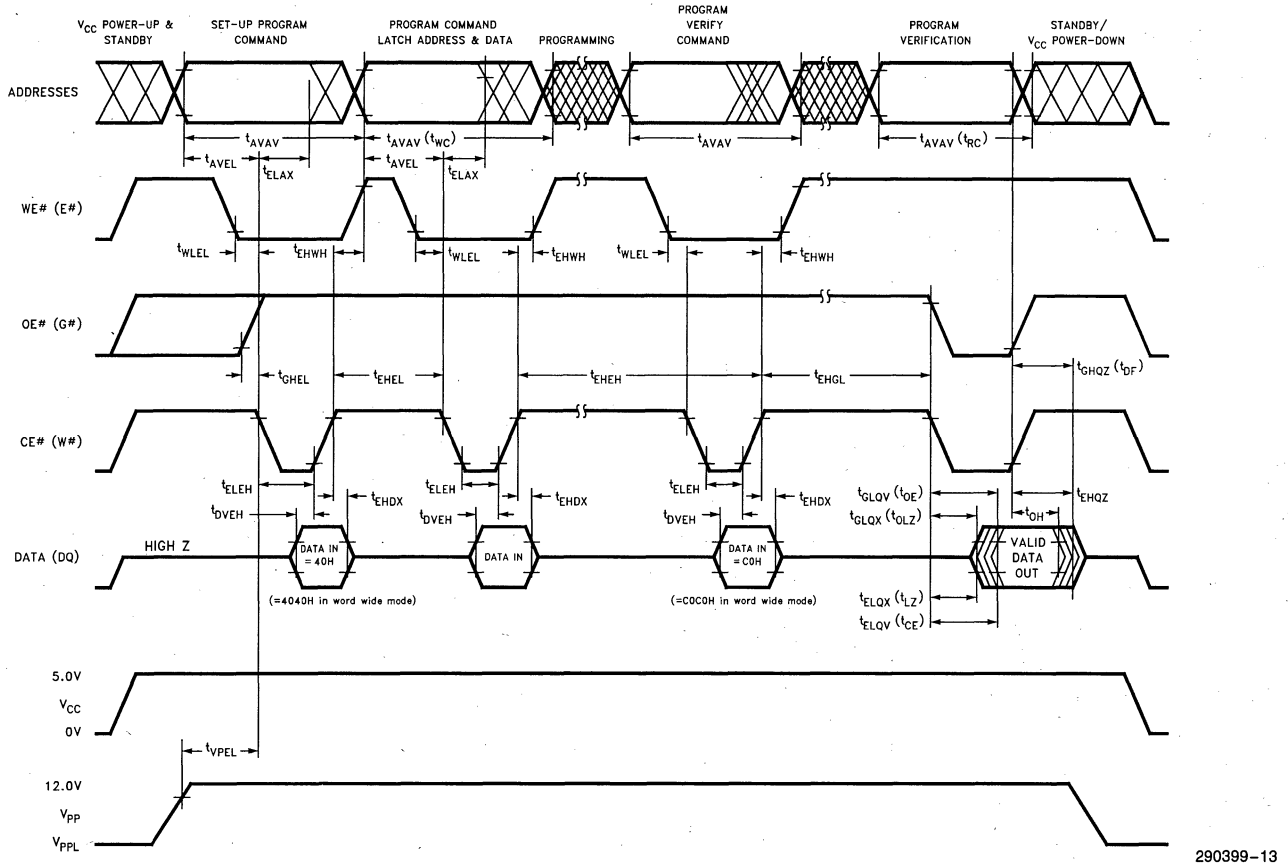
Figure 12. AC Waveforms for Erase Operations

NOTE:
CE# refers to $CE_{1,2}$#

ALTERNATIVE CE#-CONTROLLED WRITES

Symbol	Parameter	Notes	Min	Max	Unit
t_{AVAV}	Write Cycle Time		200		ns
t_{AVEL}	Address Set-Up Time		0		ns
t_{ELAX}	Address Hold Time		100		ns
t_{DVEH}	Data Set-Up Time		80		ns
t_{EHDX}	Data Hold Time		30		ns
t_{EHGL}	Write Recovery Time before Read		6		μs
t_{GHEL}	Read Recovery Time before Write		0		μs
t_{WLEL}	Write Enable Set-Up Time before Chip-Enable		0		ns
t_{EHWH}	Write Enable Hold Time		0		ns
t_{ELEH}	Write Pulse Width	1	100		ns
t_{EHEL}	Write Pulse Width High		20		ns
t_{PEL}	V_{PP} Set-Up Time to Chip-Enable Low		100		ns

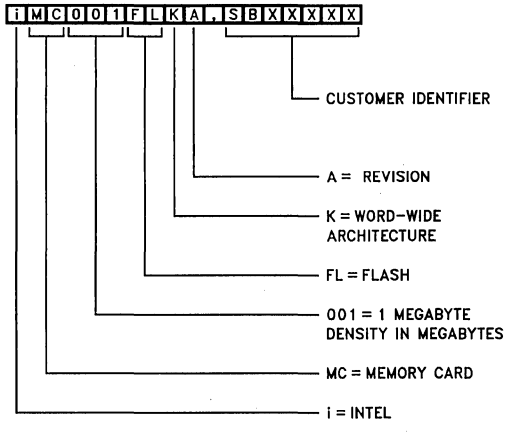
NOTES:
1. Chip Enable Controlled Writes: Write operations are driven by the valid combination of Chip Enable and Write Enable. In systems where Chip Enable defines the write pulse width (with a longer Write Enable timing waveforms) all set-up, hold and inactive Write Enable times should be measured relative to the Chip Enable waveform.



NOTE:
CE# refers to $CE_{1,2}$#

Figure 13. Alternative AC Waveforms for Write Operations

ORDERING INFORMATION



ADDITIONAL INFORMATION	Order Number
ER-20, "ETOX II Flash Memory Technology"	294005
RR-60, "ETOX II Flash Memory Reliability Data Summary"	293002
AP-343, "Solutions for High Density Applications using Flash Memory"	292079
RR70, "Flash Memory Card Reliability Data Summary"	293007

REVISION HISTORY

Number	Description
03	—Removed PRELIMINARY —Removed ExCA Compliance Section —Clarified need for Valid Address during commands —Corrected $V_{PP} = V_{PPH}$ in Erase Algorithm —Increased I_{CC2}–I_{CC5} D.C. current specs for both byte wide and word wide modes —Revised and updated Application Section discussion —Changed order number —Corrected Erase Algorithm Pulse count to 3000
04	—Change signals with bars to # —Change T/C values

6

intel®

AP-606

APPLICATION NOTE

Interchangeability of Series 1, Series 2, and Series 2+ Flash Memory Cards

SEAN CASEY
TECHNICAL MARKETING ENGINEER

November 1994

PRELIMINARY

Order Number: 292158-001

INTERCHANGEABILITY OF SERIES 1, SERIES 2, AND SERIES 2+ FLASH MEMORY CARDS

CONTENTS

PAGE

1.0 INTRODUCTION 6-172

2.0 INTEL FLASH MEMORY CARD FEATURES 6-172

2.1 Card Components 6-172

3.0 HARDWARE CONSIDERATIONS .. 6-173

3.1 Using a Series 2/2+ Card in a PCMCIA R1.0 System 6-173

3.1.1 RDY/BSY#: PIN 16 6-173

3.1.2 RESET: PIN 58 6-174

3.1.3 WAIT#: PIN 59 6-174

3.1.4 REG#: PIN 61 6-174

3.1.5 ADDRESS LINES 6-174

3.2 Using a Series 1 Card in a PCMCIA R2.1 System 6-174

3.2.1 RDY/BSY#: PIN 16 6-174

3.2.2 RESET: PIN 58 6-175

3.2.3 WAIT#: PIN 59 6-175

3.2.4 REG#: PIN 61 6-175

3.2.5 ADDRESS LINES 6-175

3.3 Series 2+: VS1 Pin 43 6-175

CONTENTS

PAGE

4.0 SOFTWARE DESIGN CONSIDERATIONS 6-175

4.1 Card Determination 6-175

4.1.1 PCMCIA R1.0: No REG# Functionality? 6-175

4.1.2 Card Information Structure (CIS) 6-176

4.2 Density Identification without a CIS 6-177

4.3 Write and Erase Algorithms 6-178

4.3.1 Pre-Conditioning 6-178

4.3.2 Write/Erase Operation Completion 6-179

4.3.3 Write/Erase Verification 6-179

4.4 Command Comparison 6-181

4.5 Series 2 to Series 2+ Software Conversion 6-181

5.0 SUMMARY 6-182

6.0 RELATED DOCUMENTS 6-183

1.0 INTRODUCTION

Many OEMs are incorporating PCMCIA-compatible card sockets into their computer systems. The benefits of using a removable memory media include easy system-to-system data transfer, rugged design, and easy density upgrades. However, this interchange capability presents the designer with the issue of ensuring system compatibility with the variety of available cards. As an example, OEMs wonder if their systems designed for the Series 2[1] or Series 2+[2] Flash Memory cards (or PCMCIA Release 2.1) will be compatible with Series 1[3] Cards and vice versa.

NOTES:

1. Series 2 Cards refer to the iMC002FLSA, iMC004FLSA, iMC010FLSA, iMC020FLSA.
2. Series 2+ Cards refer to the iMC004FLSP, iMC008FLSP, iMC020FLSP, iMC040FLSP.
3. Series 1 Cards refer to the iMC001FLKA, iMC002FLKA, iMC004FLKA.

The purpose of this document is to discuss the ability of Series 1, Series 2 and Series 2+ Flash Memory Cards to operate in systems designed using various PCMCIA Releases. From a hardware standpoint, no modifications need to be made and software compatibility may be accomplished with minimal engineering effort. This document concentrates on the differences between the three types of Intel Flash Cards, the flash memory devices within them, and how the cards operate with different PCMCIA Specification Releases.

2.0 INTEL FLASH MEMORY CARD FEATURES

To understand interchangability, let's begin with a discussion of the Series 1, Series 2, and Series 2+ Card features (refer to Table 1). All three cards contain ASICs that provide the PCMCIA-compatible electrical interface and decode and buffer circuitry for the flash memory devices. Only the Series 2 and Series 2+ Cards have the additional benefits of a Card Information Structure (CIS) and Component Management Registers (CMRs). The CIS provides the system with pertinent information about the card, such as card density, manufacturer and device identification, access speed, etc. This information allows a software interpreter to easily determine the card type inserted into the system. The CMRs supply a software-controlled interface for the flash memory devices in the card.

2.1 Card Components

At the most fundamental level, the three card types utilize different types of flash memory devices; Series 1 Cards contain ETOX™ II (28F010 and 28F020), Series 2 Cards contain ETOX™ III (28F008SA) and Series 2+ Cards contain ETOX™ IV (28F016SA) flash memory devices. The Series 1 Card memory components are classified as bulk-erase flash memory. An Erase operation erases each component in its entirety. The Series 2 and Series 2+ Cards memory components are FlashFile™ devices, separated into 64-Kbyte erasable blocks. These components have improved write and erase circuitry that perform the operations automatically after the system delivers the appropriate command. Write and erase completion can be determined by monitoring the cards RDY/BSY# output. (The detailed differences and examples of these Write and Erase operations will be discussed later in this document.) Additionally, the 28F008SA and 28F0016SA components provide a deep power-down mode that greatly reduces power consumption in battery-powered systems.

Table 1. Flash Card Feature Summary

	Series 1: iMCxxxFLKA	Series 2: iMCxxxFLSA	Series 2+: iMCxxxFLSP
Density (Megabytes)	1, 2, and 4	2, 4, 10, and 20	4, 8, 20 and 40
Flash Technology	ETOX II, 1.0 μ	ETOX III, 0.8 μ	ETOX IV, 0.6 μ
Erase Granularity	Bulk Erase: Full Chip	16–64-Kbyte Blocks Per Chip	32–64-Kbyte Blocks Per Chip
PCMCIA Compatibility	Release 1.0	Release 1.0, 2.0, 2.01 and 2.1	Release 1.0, 2.0, 2.01 and 2.1
Component Management Registers	No	Yes	Yes
Card Identification	Intelligent Identifier	Card Information Structure or Intelligent Identifier	Card Information Structure or Intelligent Identifier
Erase Operation	Requires SW or HW Timers	Automated (Uses RDY/BSY#)	Automated (Uses RDY/BSY#)
Erase Suspend	Not Applicable	Allows Read Priority	Allows Read/Write Priority
Data-Write Operation	Requires SW or HW Timers	Automated (Uses RDY/BSY#)	Automated (Uses RDY/BSY#)
Power-Down Mode	No	Yes	Yes
Power Supply Capabilities	5V V_{CC}/12V V_{PP}	5V V_{CC}/12V V_{PP}	3.3V ONLY or 5.0V ONLY or 3.3V_{CC}/12V_{PP} or 5.0V_{CC}/12V_{PP}
Individual Block Locks	No	No	Yes
Read Access Time	200 ns	200 ns	150 ns
Page Buffers	No	No	Yes

6

3.0 HARDWARE CONSIDERATIONS

In this section we will examine the hardware differences between the PCMCIA Release 1.0 (R1.0) and Release 2.1 (R2.1) specifications. In particular, this section explains the results of plugging a Series 2 or Series 2+ Card into a PCMCIA R1.0 socket and plugging a Series 1 Card into a PCMCIA R2.1 socket.

3.1 Using a Series 2/2+ Card in a PCMCIA R1.0 System

To be backwards-compatible, cards conforming to the PCMCIA R2.1 specification must function in a PCMCIA R1.0 socket. Four pin definitions were changed between PCMCIA R1.0 and R2.1. The Series 2/2+ card hardware has been designed to handle compatibility issues when plugging into a PCMCIA R1.0 socket.

3.1.1 RDY/BSY# PIN 16

Series 2 and Series 2+ Cards output a RDY/BSY# (Pin 16) signal to monitor the status of the automated write or erase algorithm. Pin 16 on a PCMCIA R1.0 socket is a No Connect on the host side, incapable of RDY/BSY# functionality. Timing delays need to be implemented in software to allow the automated write and erase algorithms to successfully complete, since polling of the RDY/BSY# signal cannot be accomplished.

3.1.2 RESET: PIN 58

The PCMCIA R2.1 specification states that "Cards requiring RESET [such as the Series 2/2+ Cards] must enter the unconfigured state each time power is applied." The RESET signal was not supported in a PCMCIAR1.0 system. To meet this requirement, the Series 2 and Series 2+ Cards contain special circuitry to internally generate a power-on reset in less than one millisecond after power is applied. This functionality ensures that a card is reset when inserted into a PCMCIA R1.0-compatible socket. Note that all new socket designs must support the RESET signal to be PCMCIA R2.1 compatible.

CAUTION:
A system designer may have tied PCMCIA R1.0 "No Connect Pins" to either V_{CC} or ground so the system outputs didn't float. Pin 58 tied to V_{CC} will cause a Series 2 or Series 2+ Card to be in a constant state of RESET.

3.1.3 WAIT#: PIN 59

The WAIT# signal is asserted by a card to delay completion of the memory-access cycle in progress. The Series 2 and Series 2+ Cards do not require this functionality; connecting the WAIT# signal to a "No-Connect" on a PCMCIA R1.0 host socket poses no problem. For PCMCIA R2.1 systems, the Series 2 and Series 2+ Cards drive the WAIT# signal high to indicate an "always-accessible" status.

3.1.4 REG#: PIN 61

The REG# signal functionality was optional for PCMCIA R1.0 socket designs and a requirement for PCMCIA R2.0 and beyond. The Component Management Registers and Card Information Structure in the attribute memory plane of the Series 2 and Series 2+ Flash Memory Cards cannot be accessed without REG# functionality.

What functionality do Series 2 and Series 2+ cards lose with the absence of REG# in a PCMCIA R1.0 socket?

Without REG#, A Series 2 CANNOT use:

- Software Reset Register
- Global Reset Power-Down Register
- Software Write Protection Register
- Software Sleep Control Register
- High Performance Ready/Busy Mode
- Tuple Information in the CIS

Without REG#, A Series 2+ Card CANNOT:

- Use All Series 2 Functionality Listed above
- Operate in 3.3V or 5.0V Only Mode
- Erase Locked Blocks

Most of the performance features and design improvements the Series 2 and Series 2+ Card offer CANNOT be fully realized in a PCMCIA R1.0 socket without REG# functionality. However, some customers may still opt to use a Series 2/2+ generation card in a PCMCIA R1.0 socket to obtain higher card densities or improved performance.

CAUTION:
Memory blocks will be locked on a Series 2+ card as a safety precaution if power is interrupted (i.e., card is removed from the socket) during a Write or Erase command. The locked blocks cannot be unlocked in a system that does not support REG# functionality. A card partition with locked blocks cannot be reformatted.

3.1.5 ADDRESS LINES

The inputs to the address line on a PCMCIA R1.0 compliant socket should not cause any problems unless. . .

CAUTION:
The upper address lines of a Series 1 card are "No Connects" on the card side. The upper address lines to the card socket may have been left floating and should not affect memory accesses to the card. The decoding circuitry on the Series 2 and Series 2+ decodes A_0–A_{25}. The upper address lines MUST be driven for the Series 2 and Series 2+ cards to prevent accesses to the wrong memory location.

3.2 Using a Series 1 Card in a PCMCIA R2.1 System

A system designed to meet PCMCIA R2.1 specifications supports the following signals which are not used on the Series 1 Card: RDY/BSY#, RESET, REG# and WAIT#.

3.2.1 RDY/BSY#: PIN 16

The flash memory devices inside a Series 1 Card do not have automated Data-Write and Block-Erase features and, therefore, do not support a RDY/BSY# interface. A card that supports the RDY/BSY# signal drives this signal high when the card is ready to be accessed.

As specified by the PCMCIA R2.1 specification, the host system must provide a pull-up resistor (50 pF, 400 μA sink, 100 μA source) on this pin to accommodate Series 1 cards that do not support the RDY/BSY# signal. For a Series 1 Card, system software must implement manual programming algorithms with timing delays to handle write and erase functionality (see Software Section 4.3).

3.2.2 RESET: PIN 58

The RESET signal, Pin 58, is input to a "No Connect" on the Series 1 Card. The RESET signal is used by some PCMCIA cards to reset any internal registers or devices to an unconfigured state. Series 1 Cards do not have any internal registers to reset and do not require RESET pin functionality. After starting a Data-Write or Device-Erase on a Series 1 Card, the host may wish to abort the operation and reset the card. The abort operation can be done in software by issuing the Reset Command (FFH) directly to the Series 1 card devices.

3.2.3 WAIT#: PIN 59

The Series 1, Series 2 and Series 2+ Cards do NOT support a WAIT-pin function. When used by other types of PC cards (such as modems or faxes), the WAIT signal is asserted by a card to delay completion of the memory-access. To accommodate cards that do not support this signal, PCMCIA R2.1 and beyond requires a pull-up on the host system's interface (50 pF, 400 μA sink, 100 μA source).

3.2.4 REG#: PIN 61

The Series 1 Flash Memory Card does not implement a PCMCIA Card Information Structure (CIS) in attribute memory, therefore eliminating the need to support a REG# pin. A Series 1 Card can be formatted with a CIS located within the memory space taken from the bottom of available Common Memory to be PCMCIA R2.01-compliant. For the simple memory array contained within the card, practically all important details of the card can be determined by using the device ID mode and a lookup table stored within the code accessing the card. (Refer to Card Type Determination section.)

3.2.5 ADDRESS LINES

The upper address lines of a PCMCIA R2.1 socket will drive "No Connects" on Series 1 cards. To prevent memory wrap-around when trying to read from a Series 1 card, the system software should not attempt to access memory beyond the card's known capacity. See Section 4.1 and 4.2 to determine the component type and density for a Series 1 card.

3.3 Series 2+: Pin 43 VS1

PCMCIA redefined Pin 43 and Pin 57 as Voltage Sense Signals (Card Outputs) to detect whether on not a card is capable of operating in a 3.3V system. PCMCIA R1.0, R2.0, and R2.1 originally defined Pin 43 as a REFRESH signal for use with PSRAM cards. Series 2+ Flash Memory Cards, which are capable of 3.3V operation, tie Pin 43 to GND to conform with the latest PCMCIA specification. VS2, Pin 57, was unused in previous PCMCIA revisions and remains unconnected for the Series 2+ card.

CAUTION:
In systems which implemented the REFRESH functionality for PSRAM cards, the Series 2+ card VS1 signal tied to GND could cause the system to lock. The REFRESH functionality may need to be disabled on the older system for use with a Series 2+ card.

4.0 SOFTWARE DESIGN CONSIDERATIONS

This section will take a look at the various software controlled functions involving Series 1, Series 2 and Series 2+ Cards. It will describe how to determine the card type and size then subsequently choose the correct software algorithms to employ for writing and erasing data (aside from differing card densities and access speeds, all three cards are virtually identical from a read standpoint).

6

4.1 Card Determination

Using the Card Information Structure (CIS), the PCMCIA specification provides a standardized method for determining a card's characteristics in a PCMCIA R2.0 and beyond socket. Although a factory-ready, built-in CIS is not required, many cards include a CIS and therefore new systems should include software to interpret the CIS.

4.1.1 PCMCIA R1.0: NO REG# FUNCTIONALITY?

Some PCMCIA R1.0 sockets may have been designed without REG# functionality since the Series 1 Card does NOT have an attribute memory plane. Software can identify the card type by issuing an Intelligent Identifier Command (9090H) and then reading the Manufacturing and Device codes. A look-up table can be referenced to identify the card type (see Table 2).

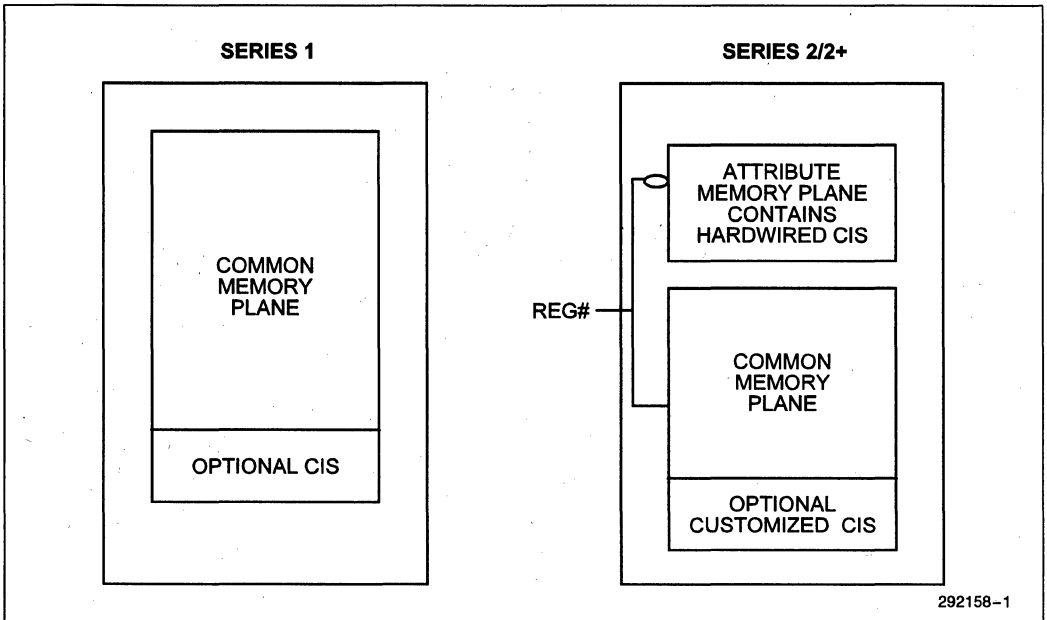
4.1.2 CARD INFORMATION STRUCTURE (CIS)

After inserting a memory card, the interpreter software should try to read the card's CIS. Using the word try indicates the possibility of finding a card without a CIS. To verify the presence of a CIS, the interpreter reads from the beginning of the CIS (Address 0000H of the Attribute Memory Plane, typically with REG# asserted) and determines a valid tuple (refer to PCMCIA spec). PCMCIA R2.1 specifies a minimal tuple requirement to meet the specification.

For example, with a Series 2 Card, the CIS begins with a 01H, indicating the possibility of a Device Information Tuple. Further processing is done to determine if that Device Information Tuple really exists or that 01H signifies random data. The Device Information will contain information about the card's speed, type and size.

According to PCMCIA, a CIS can reside in two places within a memory card (see Figure 1):

1. In the Attribute Memory Plane, accessible using the REG# pin. The Series 2 and Series 2+ Cards use the attribute memory plane implementation. The CIS contains card specific information: type of card, density, access speed, etc.
2. At the beginning of Common Memory, placed there by special format utilities provided by a system OEM or software vendor. Since a Series 1 Card does not have an Attribute Memory Plane, this method can be used for Series 1 or PCMCIA R1.0 socket implementation, but is not required.



NOTES:
Optional CIS may be added by OEMs to customize Series 2/2+
A CIS can be written into the Common Memory Plane for Series 1 cards for PCMCIA R2.1 compatibility

Figure 1. Memory Structure of Flash Memory Cards

4.2 Density Identification without a CIS

How does a system determine the card density, block count and size, etc. without using a CIS for the Series 1 card or in a PCMCIA R1.0 socket without REG# functionality? A "device ID lookup table" must be embedded in the card interpreter software. The software reads the first device pair's intelligent identifier (see Table 2), matches this value with one in the lookup table and increments the device pair counter (the card contains at least one device pair). Note that the first device pair is left in the read ID mode (see Figure 2).

After determining the component's size on the memory card by using the lookup table, increment the address pointer to the next device pair. Attempt to read the card ID without first putting it into the read ID mode. Successfully reading this device's ID indicates that the address generated a wrap-around condition and the system is back at the beginning of the memory array within the card. If the read was unsuccessful, put that device pair into Read ID Mode and perform a read to ensure a device pair is really present and increment the device-pair counter. Continue this process until every device pair in the card is accounted for. Signaled by wrapping around and encountering the first device pair which was left in read ID mode or no device pair is present. **Remember to reset all device pairs to the read array mode after determining the card density.**

Table 2. Intelligent Identifier and Block Structure

Card Type	Intelligent Identifier		Component Type	Device Count	Number of Block Pairs	Block Pair Size
	0000H	0002H				
Series 1						
1 Meg	89H	B4H	28F010	8	4	128-Kwords
2 Meg	89H	BDH	28F020	8	4	256- Kwords
4 Meg	89H	BDH	28F020	16	8	256-Kwords
Series 2						
2 Meg	89H	A2H	28F008SA	2	16	64-Kwords
4 Meg	89H	A2H	28F008SA	4	32	64-Kwords
10 Meg	89H	A2H	28F008SA	10	80	64-Kwords
20 Meg	89H	A2H	28F008SA	20	160	64- Kwords
Series 2+						
4 Meg	89H	A0H	28F016SA	2	32	64-Kwords
8 Meg	89H	A0H	28F016SA	4	64	64-Kwords
20 Meg	89H	A0H	28F016SA	10	160	64-Kwords
40 Meg	89H	A0H	DD28F032SA	10	320	64-Kwords

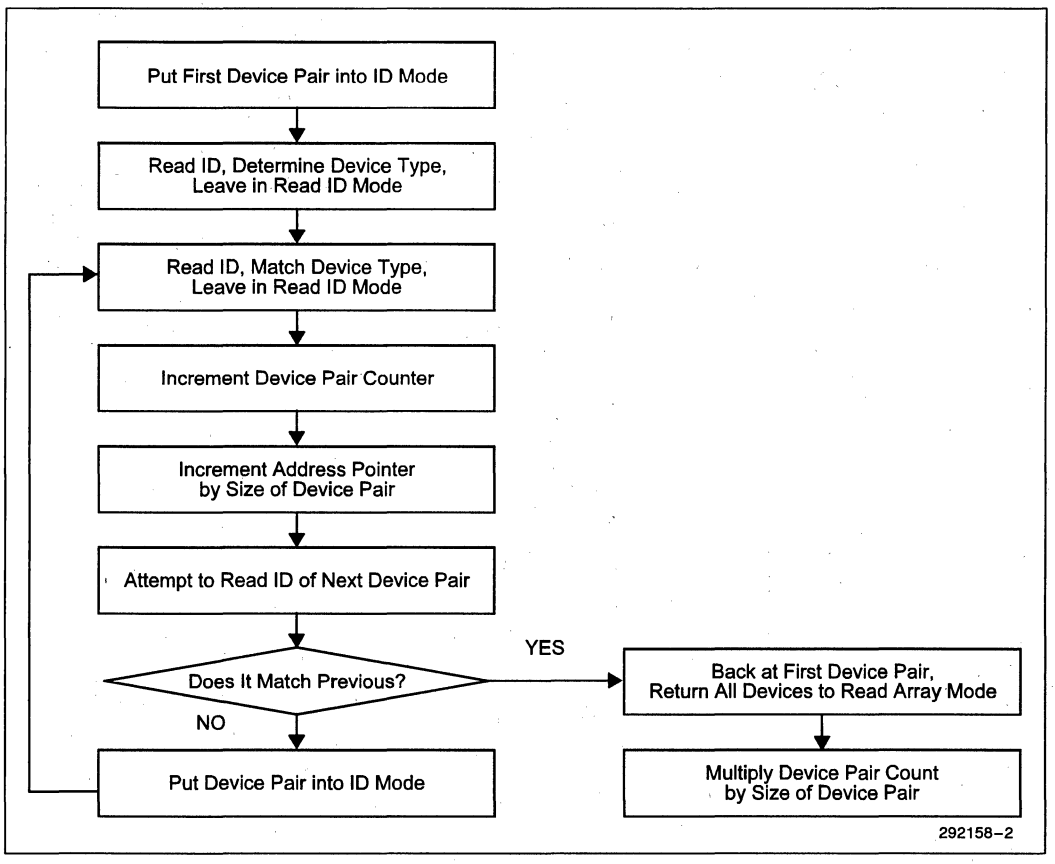


Figure 2. Card Density Determination

4.3 Write and Erase Algorithms

Similarities exist between Series 1, Series 2 and Series 2+ write and erase algorithms. For instance, to erase a device (or block), the system must first deliver an Erase Setup command followed by an Erase Confirm command. The biggest software differences exist in determining Erase and Write operation completion. Series 1 Cards require the implementation of some sort of timer; either in software or hardware. The timer determines the length of the write or erase pulse. Series 2/2+ cards have automated Write/Erase algorithms.

4.3.1 PRE-CONDITIONING

Regardless of the card type, when erasing a flash device (or block), the region being erased must first be pre-conditioned by programming all bytes to zero before the actual Erase operation begins. The flash devices in a Series 2 and Series 2+ Card automatically perform this pre-conditioning as part of the overall erase process. Therefore, Series 2/2+ card software only needs to access a single byte location of the erase block to initiate an Erase operation for the complete block.

The software requirement is significantly different for a Series 1 Card which requires manual pre-conditioning

(and access to) of every byte within the erase region. This implies, that a system using a hardware paging scheme for addressing the memory of a Series 1 Card (opposed to direct linear mapping), must switch pages (depending on the page size) to access the entire erase area.

4.3.2 WRITE/ERASE OPERATION COMPLETION

With Series 2 and Series 2+ Cards, Erase operations apply to blocks of a device rather than the entire device. (Refer to Figure 3 for an erase algorithm comparison.) The completion of the operation can be determined by polling the RDY/BSY# signal output from the card. The Series 2 and Series 2+ Cards automate the timer functions while writing or erasing, and can avoid system intervention until the system receives the ready signal from the card at the PCMCIA interface.

Series 1 Cards require software or hardware timers to monitor the completion of a write or erase cycle. In a personal computer, the erase algorithms of the Series 1 Card can become a background task by taking advantage of the system timers in lieu of timing loops. The programmer can use either the system programmable timer through Interrupt 1CH or the time-of-day clock interrupt. Using the time-of-day clock interrupt, the system intervenes to perform an erase verify and initiate another erase pulse, every 55 milliseconds (the built in stop timer of a Series 2/2+ device automatically cuts the erase pulse after 10 ms).

To demonstrate why a system designer might want to purposely decrease his system performance, assume an Erase operation requires 50 pulses (10 ms each) to complete. Excluding system overhead, the Erase operation would typically consume a total of 500 ms. In waiting for the 55 ms time-of-day clock, the flash device spends approximately 45 ms in an idle state in between erase pulses and the total Erase operation increases to 2.75 seconds. In the ideal application this number becomes insignificant because that Erase operation has occurred in the background, i.e., this erase area was "cleaning up" without impacting the user.

4.3.3 WRITE/ERASE VERIFICATION

The Series 1 memory card requires the software to perform an Erase/Write Verification after a Write/Erase command to ensure the operation has completed successfully. The Series 2/2+ memory cards have embedded the verification functionality into the Write/Erase algorithm. After a Write/Erase command has finished, a bit in the Device Status Register can be checked to see if the Write/Erase command completed successfully.

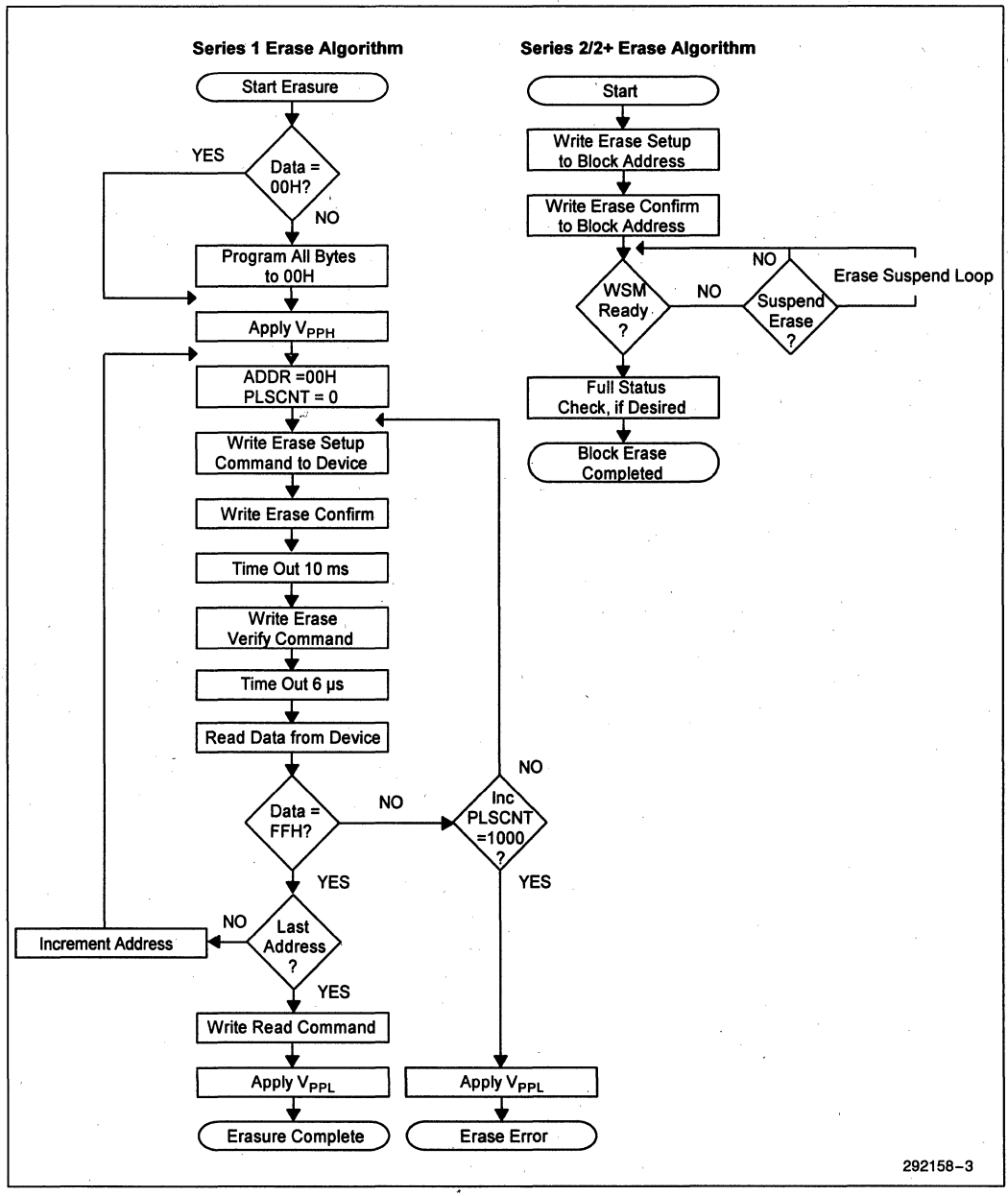


Figure 3. Series 1 versus Series 2/2+ Erase Algorithm Comparison

4.4 Command Comparison

Table 4 lists the differences in software commands used by the Series 1, Series 2 and Series 2+ Flash Memory Cards.

4.5 Series 2 to Series 2+ Software Conversion

The Series 2+ Flash Memory Card commands are a super-set to the existing Series 2 commands. A Series 2+ card is compatible with software written for a Series 2 card without any type of modification, given that the Series 2+ card can be recognized by the system.

The Series 2+ cards do have a set of "Performance Enhancement" commands (Table 3), in addition to the Series 2 compatible command set, which are used to exploit fully the enhanced feature set and performance improvement the Series 2+ card. For example: Series 2+ components have on-board page buffers which will improve the write speed to flash memory. Individual Block Locking, Page Buffer Writes, and RDY/BSY# modes are all controlled with the Series 2+ enhanced commands. Refer to the *Series 2+ Flash Memory Card User's Manual* (Order #: 297373) for a detailed description of the features of the Series 2+ Flash Memory Card.

Table 3. Series 2+ Performance Enhancement Commands

Code (H)	Series 2+ Command
08H	Page Buffer Write to Flash
71H	Read Extended Status Registers
72H	Page Buffer Swap
74H	Single Load to Page Buffer
75H	Read Page Buffer
77H	Lock Block
80H	Abort
96H	RDY/BSY# Reconfiguration
97H	Status Bits Upload
A7H	Erase All Unlocked Blocks
E0H	Sequential Load to Page Buffer
F0H	Sleep

Table 4. Software Command Comparison

	Series 1	Series 2	Series 2+ Compatible
Read Memory	00H	FFH	FFH
Intelligent ID Codes	90H	90H	90H
Set-up Erase/Erase Confirm	20H/20H	20H/D0H	20H/D0H
Erase Verify	A0H	Part of the Automated Erase Algorithm. Check Device Status Register	Part of the Automated Erase Algorithm. Check Device Status Register
Erase Suspend/Erase Resume	Issue Reset Command to Abort Erase Operation	B0H/D0H	B0H/D0H
Set-up Write/Write	40H	40H	40H
Alternate Write Set-up/Write	No Alternate Write.	10H	10H
Write Verify	C0H	Part of the Automated Write Algorithm. Check Device Status Register	Part of the Automated Write Algorithm. Check Device Status Register
Reset	FFH	FFH	FFH
Read Device Status Register	No Internal Registers	70H	70H
Clear Device Status Register	No Internal Registers	50H	50H

6

5.0 SUMMARY

This document shows that Series 1, Series 2 and Series 2+ Cards have a high degree of interchangability, given proper software and hardware design techniques. From a hardware standpoint, Series 1 Cards plug into PCMCIA R2.1 systems and Series 2/2+ Cards plug into PCMCIA R1.0 systems. Although exceptions do exist, it is strongly recommended that all new systems be designed to meet the requirements of the latest PCMCIA Version.

Software algorithm differences between the Series 1 and Series 2/2+ cards are different enough to require modifications, but a new system can easily have both algorithms embedded. The real difference is seen by the user, in the form of density options and a write performance perspective in disk emulation applications. The read performance difference will not be noticeable. However, because of higher densities, faster speeds and easier upgrade capabilities, Series 2/2+ Cards should be phased-in as the long-term production vehicle for any system requiring high performance data storage from a non-volatile PCMCIA R2.1 memory card.

Table 5. Common Problems and Troubleshooting Summary

	PCMCIA R1.0	PCMCIA R2.0
SERIES 1 iMCxxxFLKA	Does software support flash card?	Does software support flash card? No Attribute Memory Plane: No CIS No automated Write/Erase.
SERIES 2 iMCxxxFLSA	Does software support flash card? System Pin 58 (Card Reset) should be open or driven low. Is REG# (Pin 61) Functional? Upper address lines floating?	Does software support flash card?
SERIES 2+ iMCxxxFLSP	Does software support flash card? System Pin 58 (Card Reset) should be open or driven Low. Is REG# (Pin 61) functional? Upper address lines floating? Block-Locking concerns?	Does software support flash card? V_{PP} generator turned ON in single voltage system?

6.0 RELATED DOCUMENTS

Order Number	Document
290491	Series 2+ Flash Memory Card Data Sheet
290489	28F016SA, 16-Mbit FlashFile™ Memory Data Sheet
290434	Series 2 Flash Memory Card Data Sheet
290429	28F008SA, 8-Mbit FlashFile™ Memory Data Sheet
290388	Series 1 Flash Memory Card Datasheet: 4 Meg
290388	Series 1 Flash Memory Card Datasheet: 2 Meg
290245	28F020 2048K CMOS Flash Memory Datasheet
290388	Series 1 Flash Memory Card Datasheet: 1 Meg
290207	28F010 1024K CMOS Flash Memory Datasheet
292136	AB-56 "Preparing for the Next Generation Flash Memory Cards"
292095	AP-360 "28F008SA Software Drivers"
292096	AP-361 "Implementing the Integrated Registers of the Series 2 Flash Memory Card"
292099	AP-364 "28F008SA Automation and Algorithms"
292126	AP-377 "The 28F016SA Software Drivers"
292127	AP-378 "System Optimization Using the Enhanced Features"
297373	Series 2+ Flash Memory Card User's Manual
297372	28F016SA 16-Mbit FlashFile™ Memory User's Manual
	PCMCIA 1.0 Specification
	PCMCIA 2.0 Specification
	PCMCIA 2.01 Specification
	PCMCIA 2.1 Specification

intel®

AP-343

APPLICATION NOTE

Solutions for High Density Applications Using Intel Flash Memory

6

MARKUS A. LEVY
DALE ELBERT
APPLICATIONS ENGINEERING
INTEL CORPORATION

March 1994

Order Number: 292079-005

Solutions For High Density Applications Using Intel Flash Memory

CONTENTS

PAGE

INTRODUCTION 6-187

ADVANCED PACKAGING 6-187
Plastic Leaded Chip Carrier (PLCC) 6-187
Thin Small Outline Package (TSOP) 6-188
Memory Cards 6-190
Solid-State Memory Alternatives 6-190
Designing a PCMCIA/JEIDA Standard Memory Card 6-191

HARDWARE DESIGN IMPLEMENTATIONS 6-192
Design Example—A Paged-Mapped Memory Board 6-192
The Window Address 6-195
V_{PP} Generation 6-196
Page Number Selection and Reading 6-198
Optional Board Features 6-199
Switchable Data Bus Width 6-199
Linear Addressing 6-199
I/O Addressing 6-201
Capacitive Loading 6-203

CONTENTS

PAGE

SOFTWARE DESIGN IMPLEMENTATIONS 6-204
Data Recording 6-204
Interleaving 6-204
Power Requirements for Interleaving 6-208
Write-Once-Read-Many (WORM) Drives 6-208
Disk Emulation 6-209
Creating a Bootable Drive 6-212

***WHY FLASH?*—CHARACTERISTICS OF INTEL FLASH MEMORY** 6-212
Power Consumption 6-212
Power Consumption Comparison (Watts) 6-212
Data Access Time 6-212
Reliability 6-213
Weight 6-214

SUMMARY 6-214

INTRODUCTION

Mass storage encompasses many different technologies. Though commonalities exist, mass storage strives for nonvolatility, low cost per bit, and high density. Disk drives provide the best known example. However, many environments now require higher performance and reliability with lower power consumption, even at the expense of capacity. Flash memory uniquely meets these demands.

Flash memory can be used as a mass storage medium in applications including factory automation, notebook computers, high-end workstations, point of sale terminals, and data acquisition systems. Even desktop computers benefit from solid-state storage. The motivation to incorporate flash memory in any of these applications becomes obvious to the system designer who understands flash memory's benefits and density projections.

In an effort to understand these benefits, this document includes both conceptual and application oriented discussions. These discussions will be kept to a minimum with the real focus being on specific design techniques and considerations.

ADVANCED PACKAGING

Mass storage is synonymous with high density. Disk drives have increased the bit density of the rotating media via material improvements and closer tolerances. For semiconductors, density requires advanced packaging as well as higher capacity silicon (improved photolithography). Intel's Flash Memory devices are based on the company's EPROM Tunnel Oxide (ETOX) technology that enables the high degree of scaling required to achieve high density.

Intel offers the high density flash memories in several package types. The standard packages are the **P**lastic **D**ual **I**n-line **P**ackage (PDIP), the **P**lastic **L**eaded **C**hip **C**arrier (PLCC), and the **T**hin **S**mall **O**utline **P**ackage (TSOP). Advanced modular packaging in the form of PCMCIA compatible memory cards and flash drives provide the total solution.

Which package is best for your application?

Plastic Leaded Chip Carrier (PLCC)

The engineer striving to reduce board space is already using surface-mounted technology, such as PLCC. The PLCC is seen frequently on PC add-in cards and motherboards. Compared to the DIP, PLCC uses as little as 35% the overall board space. Its small size, compared to the DIP, is attributed to the terminal center-to-center spacing—50 mils versus 100 mils—as well as its four-sided pinout. No drilling or lead-cutting is necessary as leads are soldered directly to pads on the circuit board. The PLCC's 50-mil pad pitch is compatible with most circuit board manufacturing equipment. Additionally, components can be mounted on both sides of the board. However,the four-sided PLCC generally requires the use of a multi-layered board to lay out conductor traces for maximum compaction.

Thin Small Outline Package (TSOP)

When overall space constraints are critical, the TSOP is the best choice. This is best exemplified by IC memory cards. Low height is the key attribute of the TSOP, measuring 1.2 mm versus 3.5 mm for the PLCC. (Mechanical drawings in Appendix.) State-of-the-art center-to-center terminal spacing of 0.5 mm yields a smaller package and narrower conductor traces than the PLCC or DIP. In comparison, the volume of the TSOP is 172.8 mm^3 versus 656.3 mm^3 for the PLCC and 1872.3 mm^3 for the DIP.

The TSOP is available in standard and reverse pin configurations (Figure 1). Pins are located on only two ends of the package. This approach simplifies trace layout while reducing the number of board layers because traces can be routed out the non-leaded sides of the devices. Very dense board layouts are accommodated because components can literally be laid out end-to-end and side-by-side. Figure 2 displays an optimal layout best utilizing the TSOP's attributes. The close spacing allows one bypass capacitor to be used for two devices (provided they are not simultaneously selected). This optimal component layout can be mirror-imaged through the board to easily double the memory capacity.

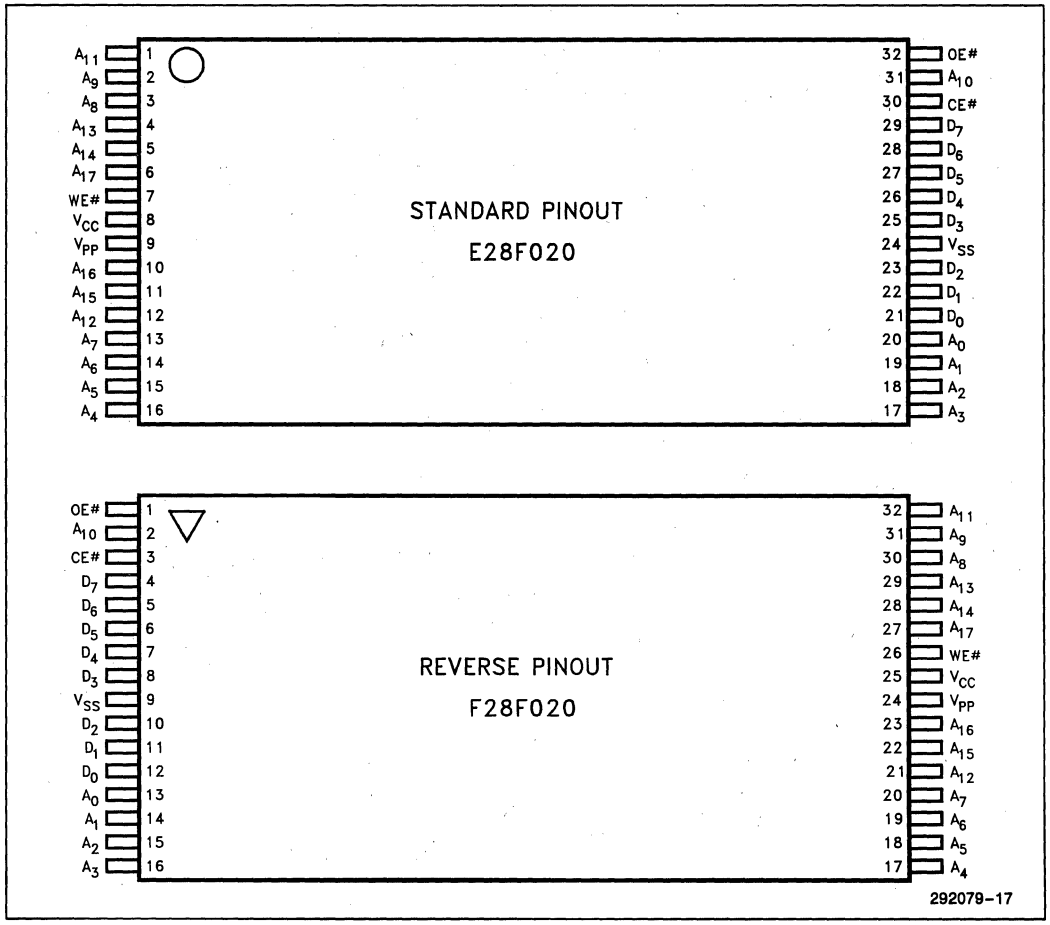


Figure 1. 28F020 32-Lead TSOP—Standard and Reverse Pinouts

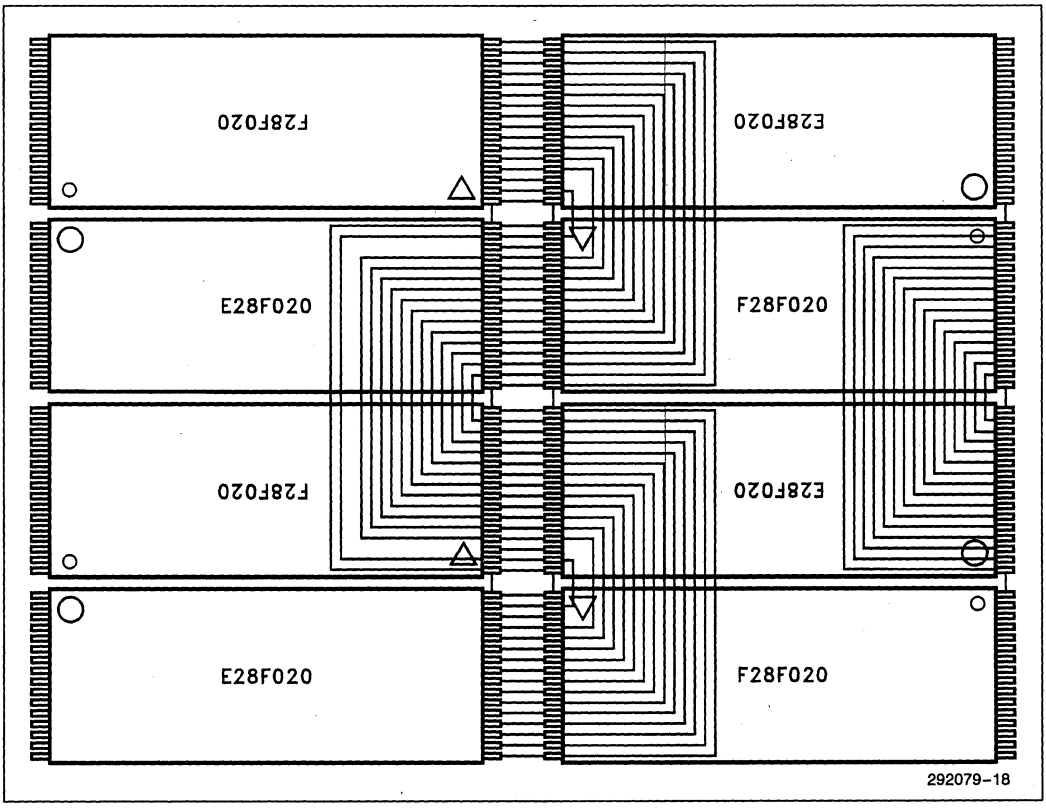


Figure 2. TSOP Optimal Layout: Highest Density Configuration (Conceptual)

6

Memory Cards

Many computer manufacturers are pursuing the IC memory card to incorporate a removable mass storage medium. This is an ideal application for the Intel Flash Memory TSOP, due to the package's minimal height.

Solid-State Memory Alternatives

ROM and SRAM are currently the dominant IC card memory technologies. ROM has the advantage of being inexpensive, but is not changeable. When newer software revisions (e.g. Lotus* 123, Windows, etc.) are available, the user must buy a new ROM card for each upgrade. Intel Flash Memory's reprogrammability minimizes the user's expense and the OEM's inventory risk.

SRAM is reprogrammable but requires batteries to maintain data, risking data loss. Like magnetic disks, flash memory is truly nonvolatile and thus has virtually infinite storage time with power off (100 years typical). Additionally, SRAM is expensive and not a high density solution. Intel Flash Memory provides a denser, more cost effective and reliable solution.

System level cost is about the same for Intel Flash Memory and SRAM + battery—

Flash memory requires 12V for programming and erasing. If a 12V supply is not available, 5V can easily be boosted. (See Application Note AP-316.) SRAM + battery requires battery state detect circuitry.

Card level cost differences are substantial (Figure 3)—

SRAM must have a battery to retain data. It also requires a V_{CC} monitor and Write Lockout circuitry. Intel's Flash Memory only requires Write Lockout circuitry (switching V_{PP} to 0V is an alternative write protect). This leads to increased area for memory components. More importantly, Intel's Flash Memory density is 4 times that of static RAM, yielding lower cost per bit.

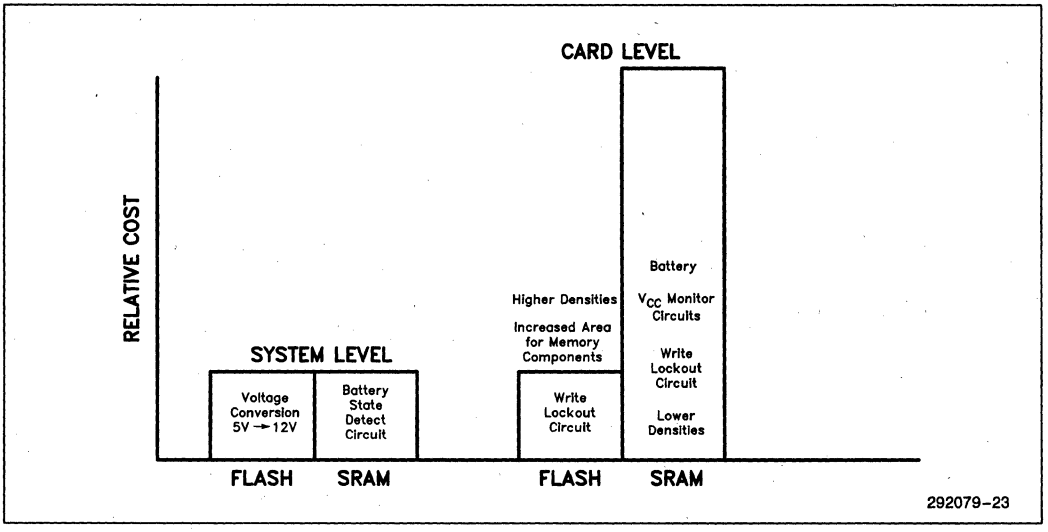


Figure 3. Support Circuitry Cost Comparison

*LOTUS is a registered trademark of LOTUS Development Corporation.
**WINDOWS is a registered trademark of Microsoft.

Designing a PCMCIA/JEIDA Standard Memory Card

Choosing among IC card design options depends on card architecture (standardization), memory capacity, data bus width, card intelligence, V_{PP} generation, and reliability.

What are the advantages of a standardized memory card pinout?

From the computer system's viewpoint, a standardized pinout enables the use of multiple third-party memory cards. This ensures competitive pricing and wide availability. From the memory card point of view, standardization allows use in a variety of systems.

The Personal Computer Memory Card International Association/Japan Electronic Industry Development Association (PCMCIA/JEIDA) 68-pin format has become the dominant IC memory card standard. Several proprietary formats are also available from their respective manufacturers, but these same manufacturers now offer PCMCIA/JEIDA versions. The PCMCIA/JEIDA standard specifies physical, electrical, information structure, and data format characteristics of the card. This standard accommodates either 8- or 16-bit system data bus widths.

The following 2 Mbyte memory card design provides a byte-addressable interface using 8-28F020s (2 Mbit, 256k x 8 devices) as shown in Figure 4. The same principles may be applied to higher density cards using higher density components. While TTL equivalent interfacing is shown, most cards will use gate arrays to reduce chip count. Address lines A18 and A19 are decoded with a 2-to-4 decoder (74HC139) to generate high and low byte chip select signals for each of the 4 pairs of flash memory devices (one pair = high and low byte). The PCMCIA/JEIDA format specifies inputs CE1# and CE2# (along with the A0 address line) select the low and high byte, respectively.

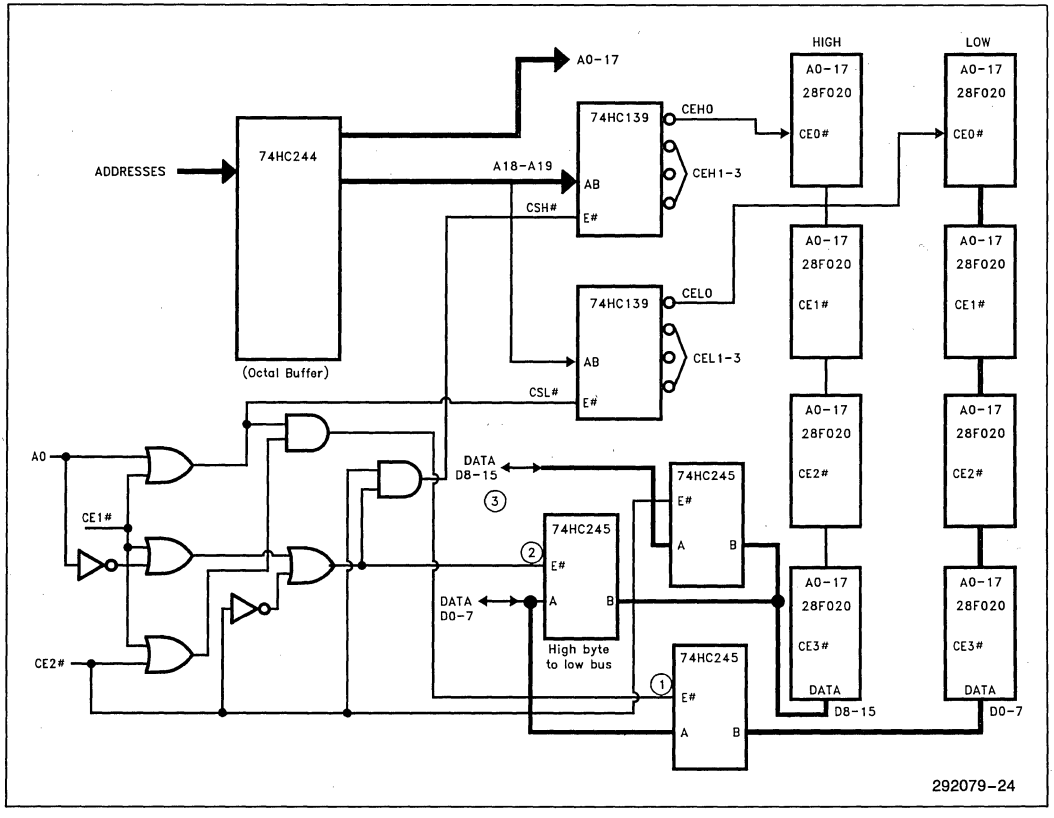


Figure 4. Decoding for PCMCIA/JEIDA Standard Bus Interface

According to the PCMCIA/JEIDA standard, the memory card is designed with the flexibility to have both an 8-bit or a 16-bit interface, dependent upon the machine it is plugged into. When the memory card is plugged into an 8-bit system, the high byte transceiver is multiplexed to the low byte of the system. In Figure 4, the highlighted transceiver (#2), maps the upper byte to the lower byte of the data bus (i.e., D_{8-15} to D_{0-7}). Signals are decoded according to the truth table in the Appendix.

One can double the memory capacity and select from among 8 pairs of flash memory devices by using a 3 to 8 decoder with inputs A_{18-20}. Notice that additional transceivers are not needed to support the additional data fanout. (See section on capacitive loading.)

HARDWARE DESIGN IMPLEMENTATIONS

Paged, linear, and I/O are the three fundamental addressing methods that can be used for accessing an array of memory devices. Linear addressing offers the fastest and most direct access to a memory array. It consumes the largest portion of the system's memory and is only practical in a 386 microprocessor (or other 32-bit processor) family system because of the large memory space available above 1 Mbyte. The I/O mapped memory array consumes the smallest amount of the system address space but has the lowest performance. A page-mapped memory array, also called a sliding AT window, is a hybrid of the linear and I/O designs. The memory array is usually very large relative to the system interface, consisting of pages typically ranging in size from 8 Kbytes to 64 Kbytes. (LIM-EMS use four to twelve 16 Kbyte pages.)

Design Example—A Paged-Mapped Memory Board

A paged design employs addressing techniques similar to the Lotus-Intel-Microsoft expanded memory specification (LIM-EMS). It allows access to one or more sections (or pages) of the flash memory array at a time. This minimal interface is particularly useful within the DOS 1 Mbyte memory space. The DOS map (Figure 5) shows 128 Kbytes of memory space available in the Optional I/O Adapter ROM area.

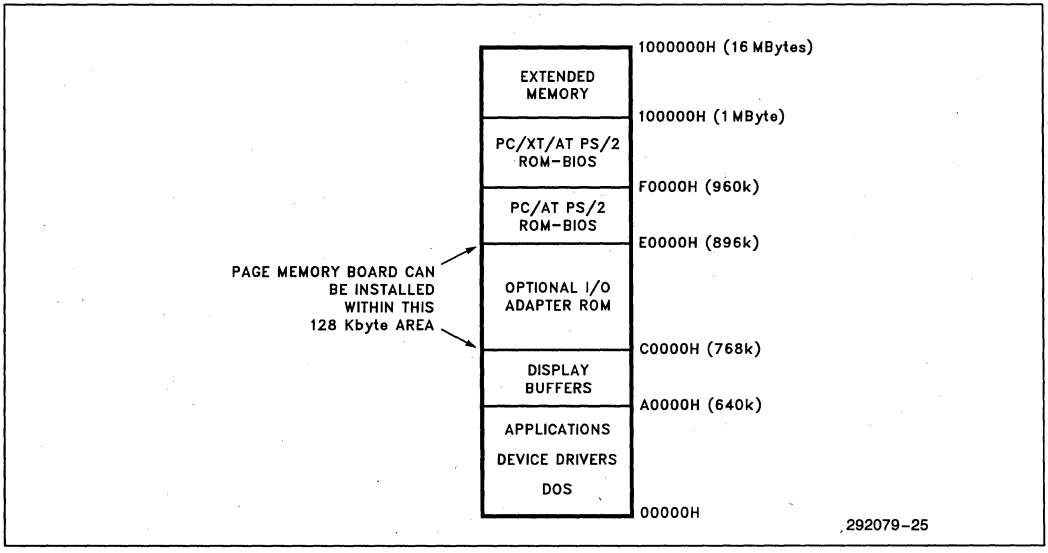


Figure 5. DOS Memory Map

Figure 6 shows the block diagram of a page-mapped flash memory board design. (Except for the addressing method, all the functional components of a board could be used on a linear or I/O mapped flash memory array.) This PC-AT*** compatible design example consists of a flash memory array and the corresponding memory and I/O decoding, V_{PP} generation, and the interface to the system bus. A page size of 64 Kbytes is used. Depending on the system's configuration, memory contention may require a smaller page size. (Note that the LIM EMS 4.0 standard uses 4 contiguous 16 Kbyte pages. Multiple pages can exist as space permits.)

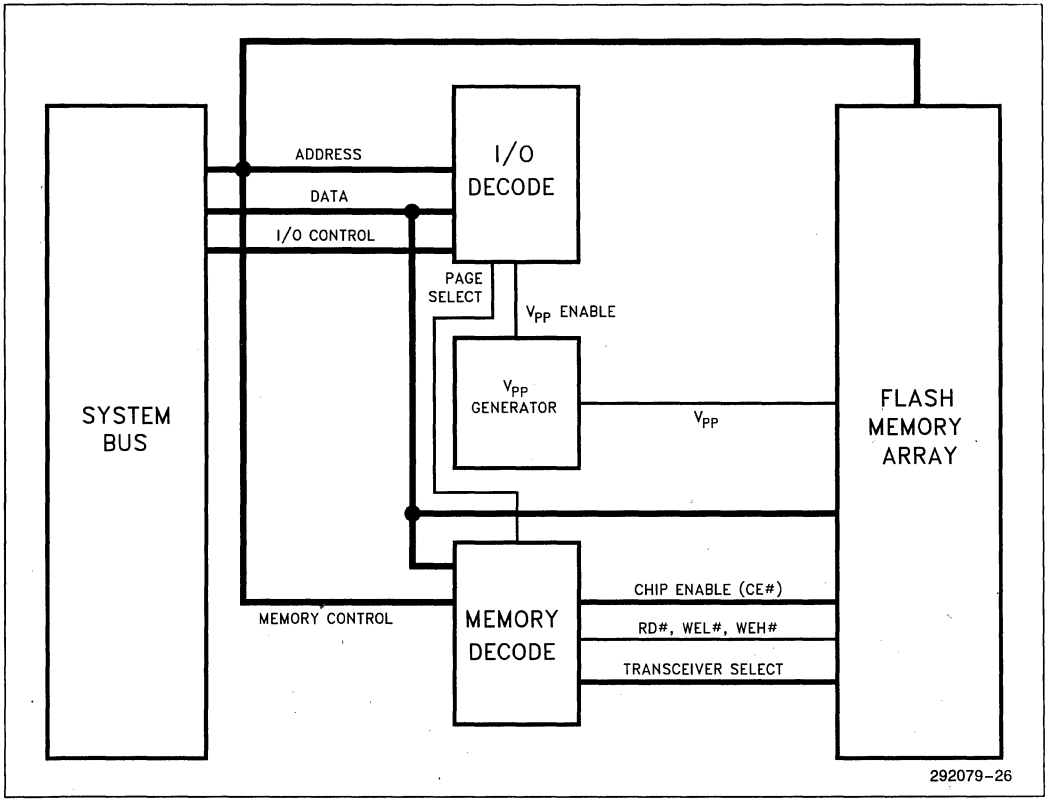


Figure 6. Page-Mapped Flash Memory Board

***PC-AT is a registered trademark of International Business Machine Corporation.

In a system design using PCMCIA/JEIDA standard memory cards the memory card is treated like a large memory array. Using a 64 Kbyte page size as an example:

Address lines A_{0-15} are supplied directly from the system address bus (after buffering). Address lines A_{16-23}, which select the pages, are sent as data to a latch before entering the memory card (Figure 7).

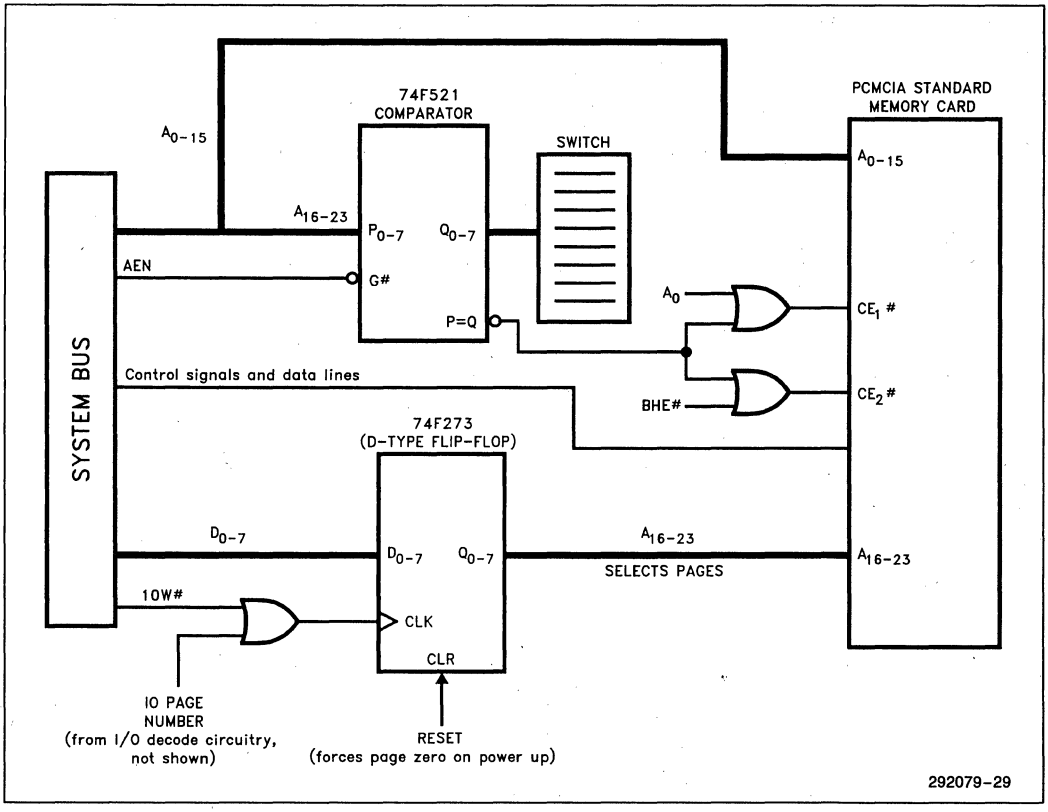
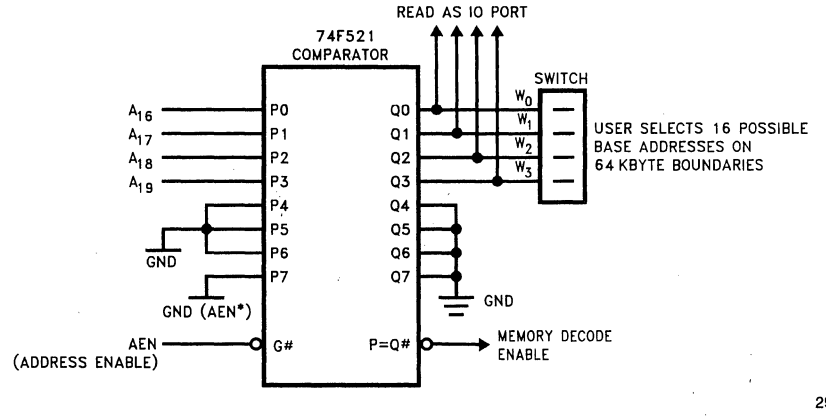


Figure 7. Memory Card Interfacing

The Window Address

A user-selectable window address can be set up on any 64K boundary below 1 Mbyte. (The memory window should be placed between C0000h and E0000h to be DOS compatible.) A DIP switch (connected to a transceiver for reading) and the four address lines A_{16-19} are the inputs to the 74F521 comparator (Figure 8). There are 16 possible window addresses. The comparator outputs the "Memory Decode Enable" signal when an address is selected that is within the 64 Kbyte window. This signal (with AEN low) allows board level memory decode.

The location of this 64 Kbyte window can be moved above 1 Mbyte by adding A_{20-23} to the comparator's inputs P_4 to P_7 of the 74F521. Bits D_{4-7} of the data bus can be connected to the comparator's pins Q_4 to Q_7 to allow reading of the full base memory address.



***NOTE:**
Ignore the contents within parentheses. This is used for section on master/slave configuration.

Figure 8. User Selects Base Memory Address

6

V_{PP} Generation

V_{PP} can be generated locally to ensure a stable, switchable 12V (±5%) supply. (Many systems generate their own 12V power supply. However, it should not be used if its regulation is greater than 5%.) On power-up, system reset, or when V_{CC} is below 4.5V, V_{PP} is forced off. It is enabled (or disabled) by writing to an I/O port address that generates the VPPEN# signal. This on/off capability is essential for battery-operated equipment and eliminates the need for WE# filtering (as discussed below). (See Intel data sheet for V_{PP} standby current.) The VPPEN# signal "ORed" with the system I/O write, IOW#, functions as the clock signal for the 74HC74 D-flip flop (Figure 9). The D-input is latched when IOW# goes high. Writing a one or a zero turns V_{PP} on or off, respectively.

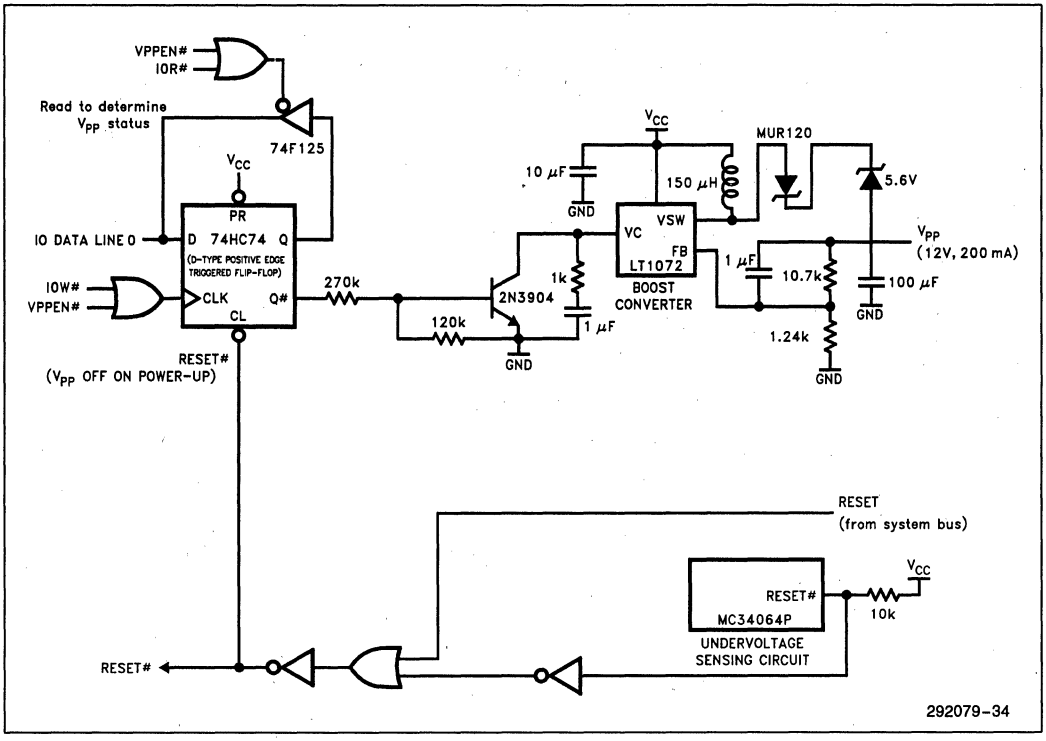
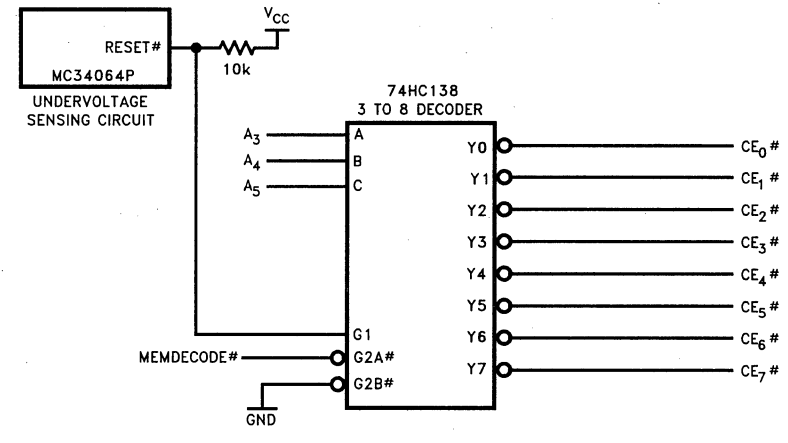


Figure 9. V_{PP} and RESET# Generation

Linear Technology's LT1072 switching regulator is used as a 5V to 12V boost converter. The FB input regulates the voltage output. The 10.7k and 1.24k resistors establish the correct reference voltage to obtain 12V. The 100 μF capacitor at the output is used to handle up to 200 mA. (See Linear Technology's LT1072 data sheet for more information.) Typically this will be much more than needed and a smaller capacitor can be used. However, this will accommodate interleaving of 8 components but may not be practical in a battery-operated system. (See section on Interleaving in the Software Design Implementation chapter.) Additionally, sufficient time should be allowed when switching V_{PP} on. The delay is a factor of the load on the line and the quality of the passive components chosen. The diode, MUR120, keeps the inductor from absorbing current from the charged output capacitor. The 5.6V zener diode ensures that when V_{PP} is less than 5.6V, the V_{PP} output is held at 0V. (This is optional if $V_{PP} \leq 5V$ is tolerable.)

During system power-up, some probability exists that noise may generate spurious writes which are actually the sequence of flash memory commands that initiate erasure or programming. Power-up protection in this design is provided by disabling V_{PP} until voltages have stabilized. The Motorola component, MC34064P, is an undervoltage sensing circuit that begins functioning when V_{CC} is above 1V. Between 1V and 4.6V, the RESET# output is active. The RESET# output or a system RESET clears the 74HC74, keeping V_{PP} off when V_{CC} is less than 4.6V. Alternatively, this signal, or a supply's "POWERGOOD" signal, may gate WE# or CE#, as is common with battery-backed SRAM or EEPROM designs. As an example, the RESET# output of the MC34064P can be tied to the active-high enable of the decoder to disable any CEs# until V_{CC} = 4.6V, as shown in Figure 10.



Chip Enables will not be active until V_{CC} = 4.6V.
At this point, signals are stable and involuntary writes will not occur.

Figure 10. Protecting the Circuit from Involuntary Erasure and Programming. Use an Undervoltage Sensing Circuit, or a System's "POWERGOOD" Signal, to Control Chip Enables

Latching a one into the 74HC74 D-input puts a zero on the output Q#.This turns off the transistor 2N3904. When the 2N3904 is off, the VC input of the LT1072 is 5V and the VOLTAGE SWITCH (VSW) output generates 12V.

Page Number Selection and Reading

It is standard practice to use an I/O port to generate the page number for this type of memory array. The potential number of pages that can be selected is determined by the size of the data bus as well as the amount of decoding the system can practically handle. In this design, this I/O port allows selection of 256 64-Kbyte pages, for a total of 16 Mbytes of flash memory. The page number is written to the 74F273, Octal D-Type Flip-Flop (Figure 11). It is latched by the rising edge clock signal derived by the "ORing" of the corresponding 74F138 decode signal (I/O PAGE NUMBER) and the system IOW#.

Page zero is automatically selected on power-up because the 74F273 clear input is connected to RESET# (generated as part of the V_{PP} circuitry). This feature ensures that the board will power up in page zero. Given the proper software, this board can be turned into the system's bootable drive. (See section on Software Design Implementations.)

The current page number can be obtained by reading the same I/O port. The I/O decoder output, I/O PAGE NUMBER, "ORed" with the system IOR#, produces the signal enabling the 74F245 bus transceiver (that is tied to the output of the 74F273).

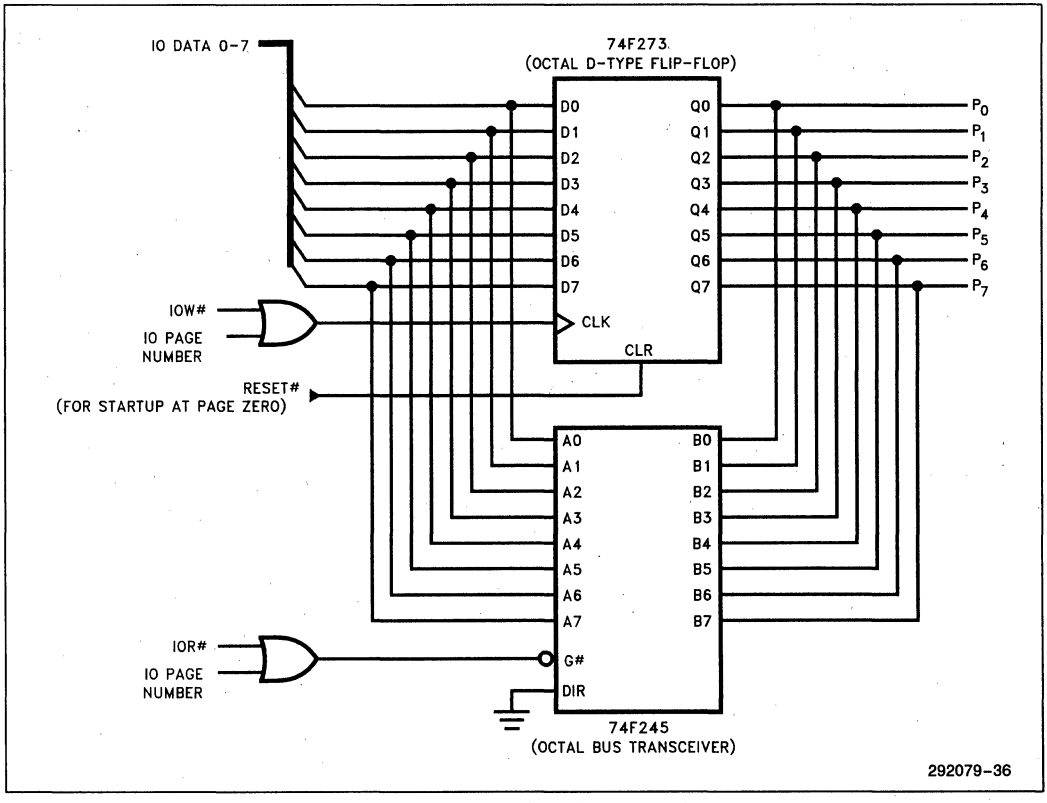


Figure 11. Selecting or Reading Page Number

Optional Board Features

So far we have described the components required to design a functional flash memory array. Optional features can be added to make an implementation more versatile in an application environment:

Switchable Data Bus Width

This feature allows a board to execute in a PC XT* (8-bit bus) or a PC AT system (16-bit bus). Memory card designs for adopting the PCMCIA/JEIDA format must include similar provisions as shown earlier. At the PC-I/O channel interface, (for use in an 8-bit system), an extra transceiver redirects the upper data bus (D_{8-15}) to the lower data bus (Figure 12). The 16BIT # signal is generated from a ground on the PC AT I/O channel extension; it will be high (because of the pull-up resistor) when a PC XT is used.

Linear Addressing

Linear addressing directly maps the flash memory array into the system's memory space. "Instantaneous

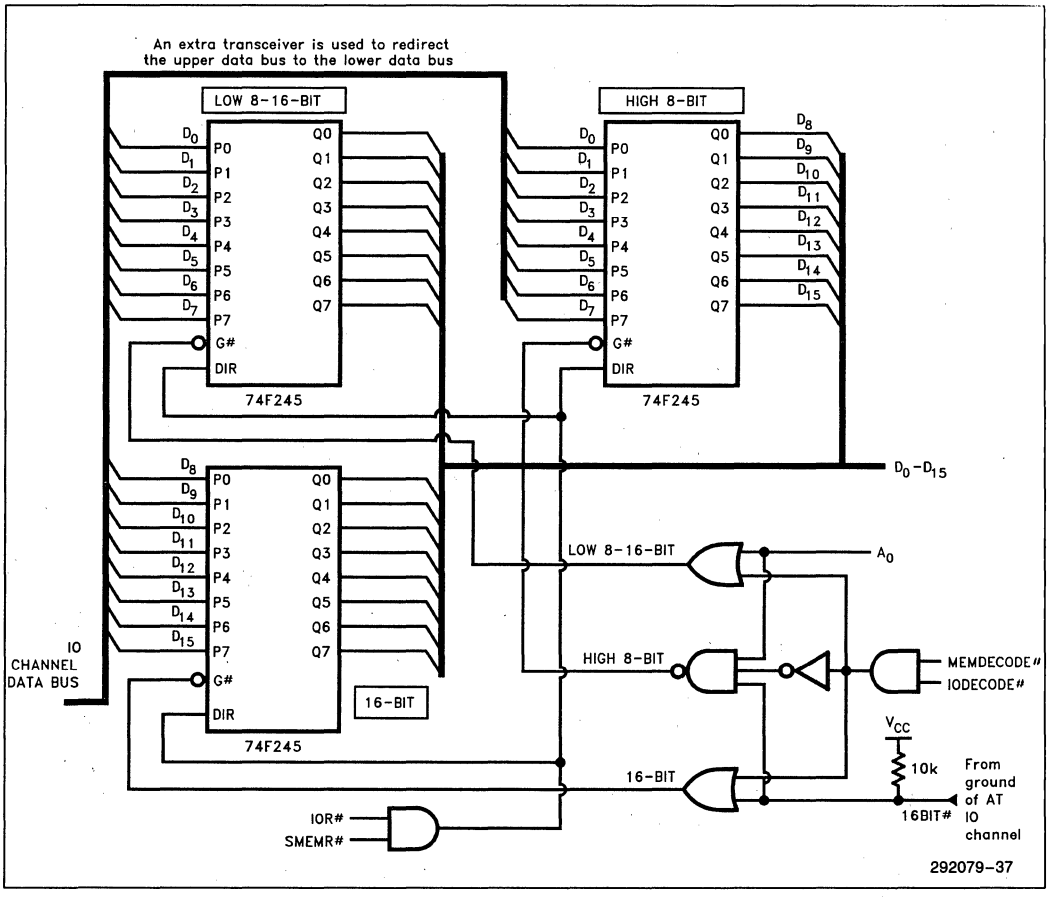


Figure 12. I/O Channel Transceiver Interface for 8- or 16-Bit Data Bus Selection

*PCXT is a registered trademark of International Business Machine Corporation.

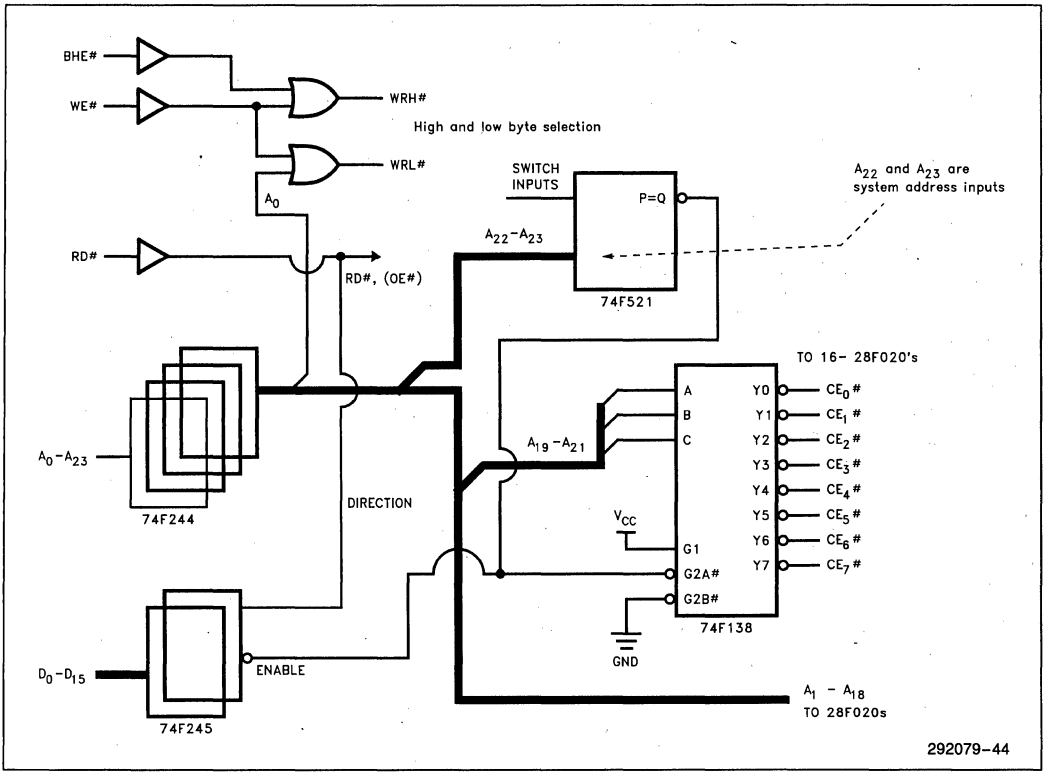


Figure 13. Linear Addressing Hardware Block Diagram

Access" of the entire array is the obvious advantage over paging. Additionally, the decode circuitry is simplified. Figure 13 shows an example for accessing 16 Intel Flash Memory 28F020s arranged in a 4 Mbyte linear array.

The number of address lines used, as well as the decoder type (2 to 4, 3 to 8, etc.), is determined by the flash memory device size. The address lines A_1–A_{18} are used for byte selection within each device (256 Kbytes * 8).

The decodes for the individual devices can be designed in a row-column method similar to that used for the page memory board. An alternative design uses an individual chip enable for each of the 16 devices.

The enable for the 74HC138 (3 to 8 decoder) is governed by a 74F521 comparator. System address inputs to the comparator are chosen to locate this array on a 4 Mbyte boundary. (The array base address could be located on a non-4 Mbyte boundary but this would add to the decoding complexity.) With the inputs chosen in this example (A_{22}–A_{23}), the array base address will be between address 0 and 12 Mbytes to confine this memory array within the PC AT defined address space of 16 Mbytes. A_{19}–A_{21} are inputs to the decoder which generates one of the eight chip enables (CE#). (Use a 74F245 transceiver for the data bus of every 8 flash memory devices. The address lines also need buffering when connected to a PC bus.)

I/O Addressing

From the standpoint of the system's address space usage, I/O addressing provides a conservative solution. As an example, four gigabytes of a flash memory array can be addressed through only two I/O ports. An I/O write sends the flash memory addresses out on the data bus. This "data" is latched (using '574s) and made available to the flash memory devices and decoding circuitry (Figure 14). A third I/O port, used as an enable for the flash memory device decoder and transceivers, helps conserve power when the array is not being accessed.

Relative to linear addressing, I/O addressing generally has limited access speed capability because of the I/O "bottleneck". Read speed can be increased to match linear addressing by replacing the '574 latches with '191 counters.

In the following circuit example, decoding for I/O is accomplshed with a 74F138, 3 to 8 decoder (Figure 15, U1). The base address for these I/O ports is on an 8-byte boundary. When any one of the 8 I/O addresses is selected, the comparator (U2) generates the enable signal (if AEN is low) for the decoder.

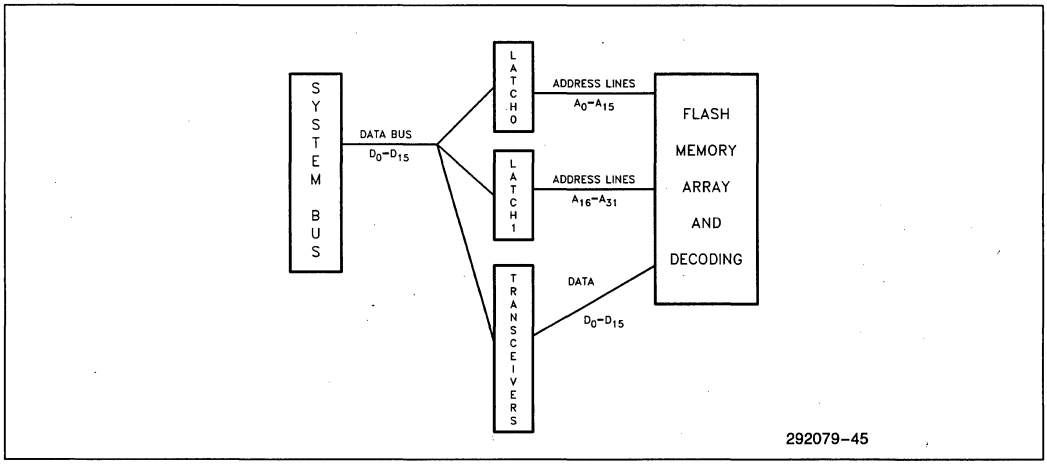


Figure 14. Data Bus Generates Flash Memory Addresses

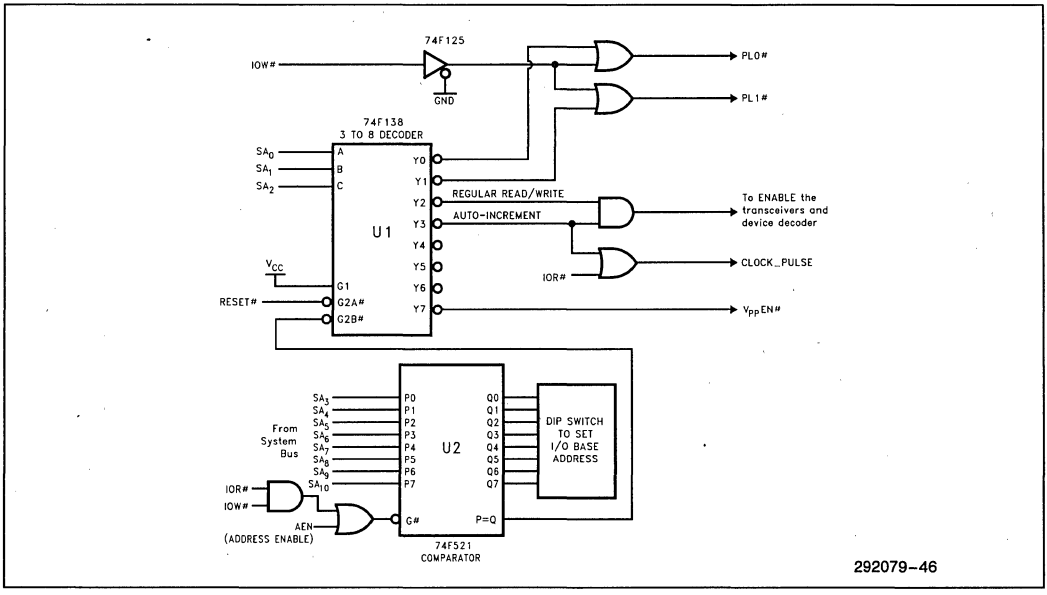
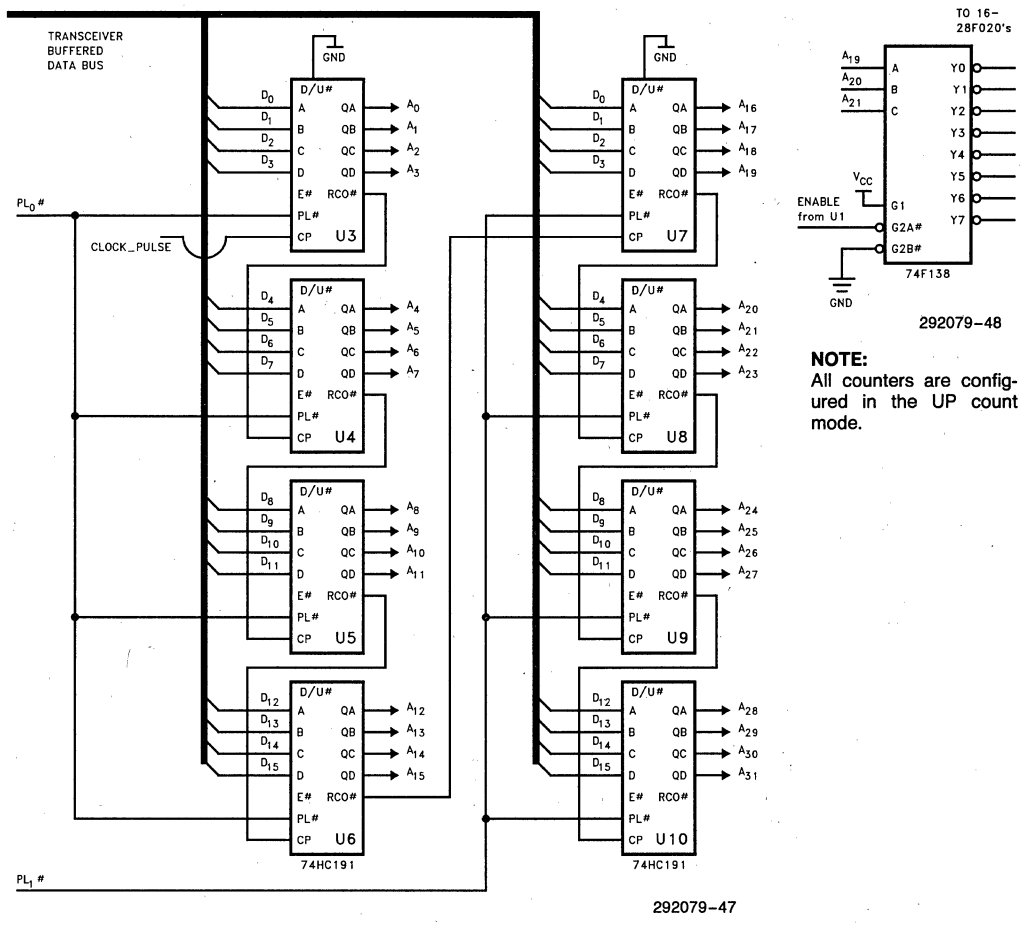


Figure 15. I/O Decode and Enable Circuitry

An I/O write to the first and second ports generates parallel load signals, PL_0# and PL_1#. These signals latch the "data" (addresses) into the 4-bit counters (Figure 16, U3–U10). This latched data represents the address for the flash memory devices.

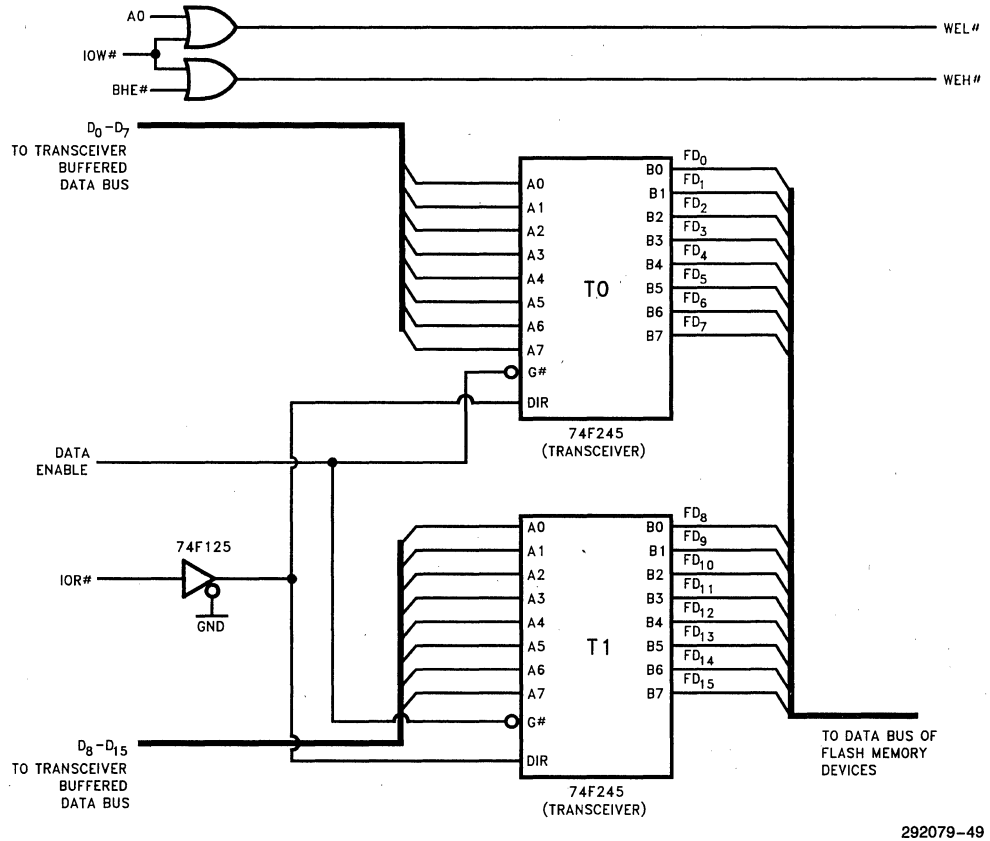
A read or write from the selected flash memory address is performed when the third I/O port is accessed (Figure 15, U1); this generates an enable for the flash memory device decoder and associated transceivers (Figure 17, T_0 and T_1).



NOTE:
All counters are configured in the UP count mode.

NOTE:
A_0–A_{31} are inputs to flash memory devices. Only address lines A_0–A_{18} are used for the 28F020s.

Figure 16. Counter Circuitry



NOTE:
The 16-28F020's are arranged as a 16-bit word configuration. WEL# and WEH# are for the low and high bytes, respectively.

Figure 17. Transceiver Enable Circuitry

The fourth I/O port activates the circuitry that obtains very high performance from an I/O board. A read from the fourth I/O port address generates the clock signal for the 74HC191s, CLOCK__PULSE. The counter increments on the rising edge of the clock (read signal), selecting the next flash memory address. This rising edge occurs at the end of the I/O read cycle and the data has already been read. This method is analogous to address pipelining. It is perfect for a "string" read because continuous reads from the fourth I/O port automatically increments the address to access the next word of data stored in the flash memory array.

Capacitive Loading

Capacitive loading is an important consideration for a solid-state mass storage device. If proper buffering techniques are not followed, performance degradation will occur.

The specifications for Intel's Flash Memory devices are based on a test capacitive load of 100 pF. Each data line contributes 12 pF, therefore 8 devices connected to one data transceiver will not experience speed derating (12 pF * 8 = 96 pF). Additional flash memory devices

on that transceiver will increase the loading seen by any one device.

Degradation is calculated as follows (Q = Amount of Charge, T = Time, C = Capacitance, V = Voltage, and I = Current):

COULOMBS LAW STATES:

$$Q = I\Delta T$$

AND GIVEN THE RELATION:

$$V = \Delta Q/C \longrightarrow I = C\,\Delta V/\Delta T$$

FROM THIS RELATION, THE CHANGE IN ACCESS TIME CAN BE EXPRESSED IN TERMS OF CAPACITIVE LOAD:

$$\Delta T = C\,\Delta V/I$$

For example, using four SIMMs, each with 8 components in a 16-bit configuration (4 components on high byte and 4 components on low byte), each Intel Flash Memory device sees a load of 15 devices (12 pF * 15 = 180 pF). This loading is 80 pF in excess of the device specification so therefore:

$$\text{Time Change} = \text{Additional Capacitance} \times \frac{(V_{CC} - V_{OL})}{I_{OL}}$$

$$= 80\text{ pF} \times \frac{(5.0 - 0.4)\text{V}}{5.8\text{ mA}} = 64\text{ ns}$$

(Reflecting worst case conditions.)

SOFTWARE DESIGN IMPLEMENTATIONS

Each hardware implementation discussed above can be used in several types of mass storage applications. The general categories include: data recoders, Write-Once-Read-Many (WORM) drives for storing application programs and fixed data, and magnetic disk emulators.

Data Recording

The applications for data recording represent an endless list. Examples include digital imaging, digital photography, point-of-sale terminals, patient monitors, and flight recorders. These systems will use Intel Flash Memory as a more economical and reliable replacement for SRAM + battery. Alternatively, mechanical disks will also be replaced by Intel's Flash Memory when higher reliability, lower power consumption, higher performance, and lighter weight are required.

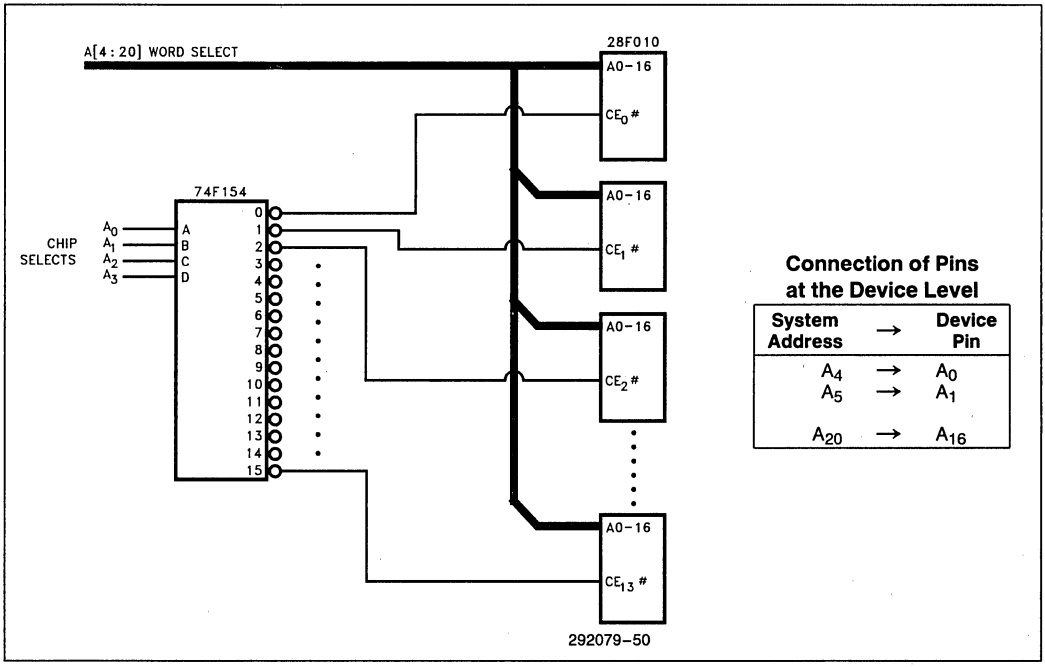
Interleaving

Although the basic concept of data recording is similar from system to system, variations in implementation exist. For instance, some applications require high-speed data acquisition. Data programming rates are improved considerably by employing interleaving techniques. The majority of time spent programming or erasing a flash memory device results from the delay times in the software algorithms. (It is advised to review the standard algorithms first. See any Intel Flash Memory data sheet for Quick-Pulse Programming algorithm.) Interleaving takes advantage of these delay times to begin programming consecutive devices.

There are hardware and software mechanisms for interleaving. The flash memory array for hardware interleaving requires special decoding techniques (Figure 18). Contrary to linear decoding, the system address lines A_0–A_3 are decoded to provide the chip select signals and individual bytes are selected with the address lines A_4–A_{20}. (For the Intel 28F010.) This decoding technique allows software to automatically access sequential devices by writing or reading sequential memory addresses. (Data accumulated with program interleaving will not be stored consecutively within a single device.)

The interleaving algorithm to program the 2 Mbyte flash memory array is shown in Figure 20 and 21. The basic goal is to utilize the delay times. To simplfy the algorithm for this discussion, the data will be programmed on a byte-wide basis. Word-wide and double word-wide techniques, discussed later, will further increase programming speeds.

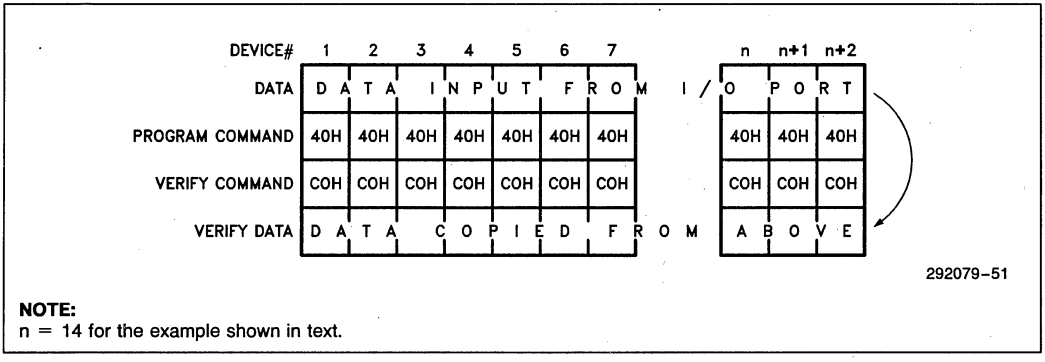
During multi-component programming, the number of pulses required could vary between different devices. Code is reduced if the programming loop does not have to selectively "decide" if a byte has programmed correctly (verified). However, continual programming of a programmed byte is not necessary and should be avoided. This is done by masking the command sent to that particular device. The RAM table in Figure 19 is used as a data and flash memory command buffer. After a programmed byte has verified, its associated data and commands in the RAM table are written with the value 0FFH (RESET command for Intel flash memory). The data is also written as an 0FFH since this is null program data.



Connection of Pins at the Device Level

System Address	→	Device Pin
A_4	→	A_0
A_5	→	A_1
A_{20}	→	A_{16}

Figure 18. Hardware Interleaving Block Diagram



NOTE:
n = 14 for the example shown in text.

Figure 19. RAM Array Used as Data Buffer and Command Mask Storage

6

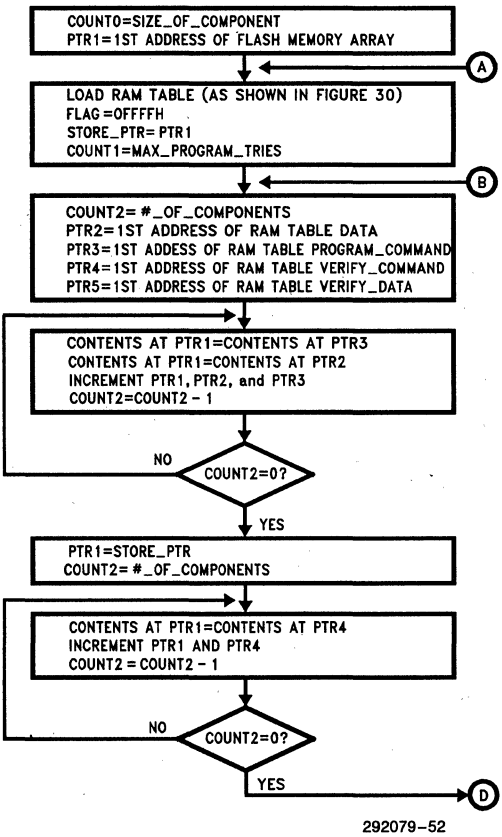
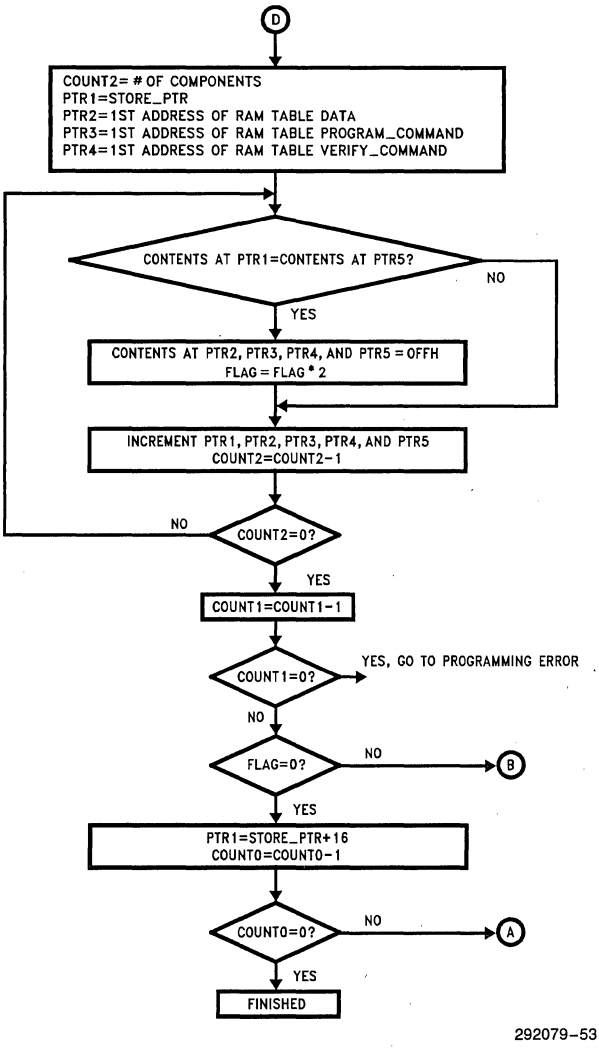


Figure 20. Program Interleaving Algorithm



Does Flash Memory Byte = Verify Byte?

Shift bits of flag left.

Exceeded program tries, go to error.

All 1's shifted out?

To access next byte of each device.

292079-53

Figure 21. Program Interleaving Algorithm (Continued)

Software and hardware interleaving are very similar. Software interleaving is performed using conventional decoding and addressing methods. Instead of incrementing flash memory addresses by one to access the next byte (as with hardware decoding), increment the address by the size of the component. While allowing the use of "general-purpose" (non-interleaved) hardware, software interleaving requires reading back the data in the same, non-sequential fashion as was used for recording.

Interleaved erase is useful for quickly erasing an array of flash memory devices. This approach greatly reduces the total subsystem format time. As specified in the erase algorithm, each erase pulse requires a 10 ms delay. (See Quick-Erase algorithm in Intel Flash Memory data sheet.) Without interleaving, the processor idles during this delay time. As with program interleaving, this time is used to begin the erasure of consecutive devices, thereby reducing the overall erase time.

Further program and erase time can be saved by supplementing the byte-wide algorithm with 16- or 32-bit interleaving. Extra data and commands are added to the RAM Mask Table. The major difference in the algorithms involves the verify operation. Depending on the bus width, 2 or 4 bytes are verified simultaneously as shown in Figure 22 (for a 16-bit algorithm).

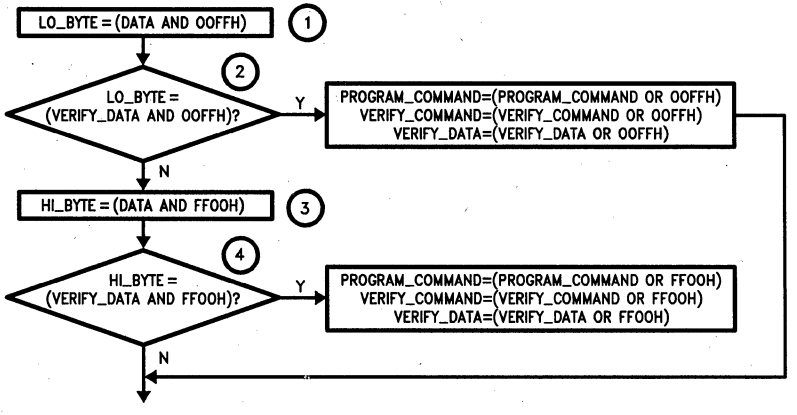
Power Requirements for Interleaving

Current consumption is an important consideration for interleaving. During programming, each device typically consumes 9 mA (1 mA I_{CC} and 8 mA I_{PP}) while programming or erasing; this translates to about 100 mW. If interleaving with 16 devices, about 144 mA (16 devices * 9 mA) or 1.6W, is drawn. Battery powered systems will have a practical limit on the number of components in the interleaving loop. Failure to accommodate these current levels, resulting in V_{PP} voltage drop, will compromise programming and erase reliability.

Write-Once-Read-Many (WORM) Drives

The optical disk is an example of a typical WORM drive application. Its strengths are extremely high densities and low cost per bit. However, it is an unacceptable solution for a low powered, lightweight laptop computer system. It is this environment that solid-state drives offer the greatest benefit. Solid-state ROMs have historically been used in portable systems to store software programs that seldom change. When the software changes, discard the ROM "application hardfile" and program a new one.

Unlike the ROM drive, Intel Flash Memories can be reused and reprogrammed in a true WORM fashion. A computer user can load favorite software programs on the flash memory drive. Add revised programs to the drive by writing to the next free space or by erasing and reprogramming the entire drive. Software drivers can be written to implement this functionality in most operating systems.



NOTES:
1. MASK the HI Byte with 00H.
2. If the LO Byte verifies, then mask the data, program, and verify command with 00FFH (RESET).
3. Mask the LO Byte with 00H.
4. If HI Byte verifies, then mask the data, program, and verify command with 00FFH (RESET).

Figure 22. 16-Bit Masking for Verify Operation

Disk Emulation

Microsoft has a flash memory file system. It stores and retrieves data or application programs in a manner that, to the end user, appears similar to a disk drive. New files are written sequentially from beginning of memory. However, when the disk is full, it reclaims memory space for storing additional files.

When an application accesses a disk through INT 21H, the MS-DOS* kernel checks the drive letter (Figure 23). If the drive has been declared as a flash memory disk, a built-in redirector services the call (analogous to networked drive accesses). Otherwise, if the drive letter is that of a floppy or hard disk, the call is handled by the standard DOS file system. The File System provides the link between DOS and the Flash Memory and Hardware device driver. It changes DOS file system commands into a form understood by this unique file structure.

The Flash File System Driver contains the "intelligence" of this file system. It searches for:

1. A Boot Record that identifies the file system and version, and locates the start of the data area;
2. The Root Directory Entry Record and many Directory and File Entry Records.

The file system driver is independent of the hardware interface to the flash memory disk. The PCMCIA device drivers, developed by the OEM or BIOS software vendors, interfaces the flash memory disk to the flash file system. The actual implementation of the interface is dependent on the hardware configuration of the disk (I/O, paged, and linear addressing are examples).

To minimize fragmentation losses and allow arbitrary extension of files, the flash memory file system uses variable sized blocks rather than the standard sector/cluster method of more traditional file systems. The

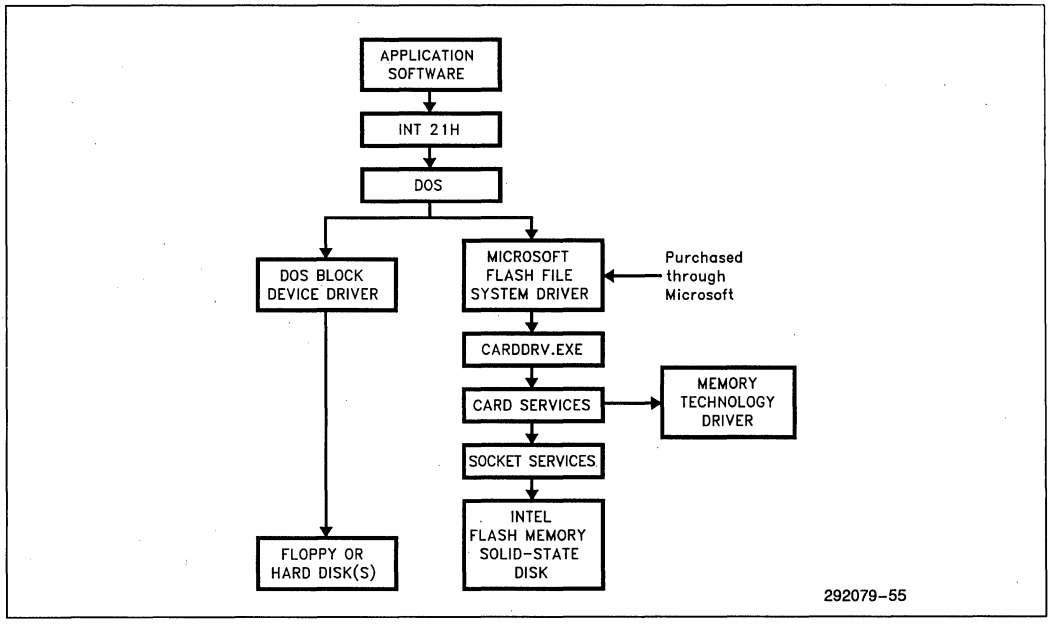


Figure 23. Disk Interface Levels

*MS-DOS and Microsoft are registered trademarks of Microsoft Corporation.

fundamental structure employed to offer this flexibility is based on linked list concepts; files are chained together using address pointers located within directory entries for each file.

Files and directories are written to the flash memory disk using sequentially free memory locations—a stack-like operation (Figure 24). Furthermore, file sizes can be variable, abandoning the traditional sector/cluster approach of DOS. When "the stack" fills up, (containing deleted files), the intelligent software algorithm performs a cleanup operation to reclaim the "dirty" space.

File and subdirectory information is essentially attached to the beginning of each file, unlike the standard DOS approach of directory and FAT placement. As directory and file entries are added, they are located by building a linked-list. Besides containing the customary fields (e.g., name, extension, time, date of creation, etc.), a directory and file entry contains a status byte and various pointers used for the linked-list process. The status byte, besides indicating whether a file/subdirectory exists or is deleted, is also used to signify valid sibling and/or child pointers and to determine if a directory entry pertains to a file or a directory.

When a directory or file is requested or added, the flash memory disk is searched beginning at the head of the linked-list. The chain is followed from pointer to pointer until the correct entry is found. If the search arrives at the chain's end (an FNULL is encountered), the system responds analogously to DOS with a "File not found" message.

This linked-list chain consists of two basic types of pointers: sibling and child. Sibling pointers are used to locate directories or files at the same hierarchial level. Child pointers are used to locate subdirectories or the first file of a particular directory. The following examples elaborate these concepts.

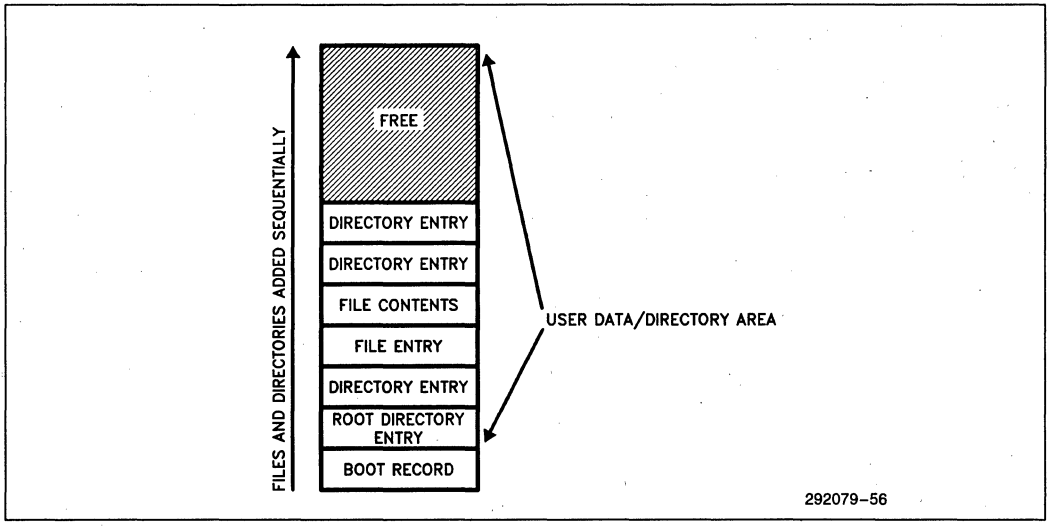


Figure 24. FFS Storage

In Figure 25, Directories B and C are subdirectories of Directory A. Specifically, Directory C is a sibling of Directory B and both are children of Directory A. FNULL indicates the end of the chain.

Figure 26 shows two files (File A and File B) added to a directory (Directory A). File A and File B are at the same level, therefore they are siblings. A file's file entry contains an extent location pointer that indicates the start of its data area.

When a file appears multiple times (because of deleted versions) on the flash memory disk, the file system must find the most recent version. The status byte contains bit fields that indicate whether that particular file is a valid or deleted file. The directory information of a deleted file is used for pointers of the linked list and the search would proceed until the most recent version is found.

A key point to be made for using this method of file storage is that the user is in control of the rate in which the disk becomes full; using the flash memory disk predominantly for application code storage and non-temporary data files reduces the frequency of "cleanup". However, flash memory will typically perform 100,000 cycles and eliminates reliability concerns when used as a hard or floppy disk replacement.

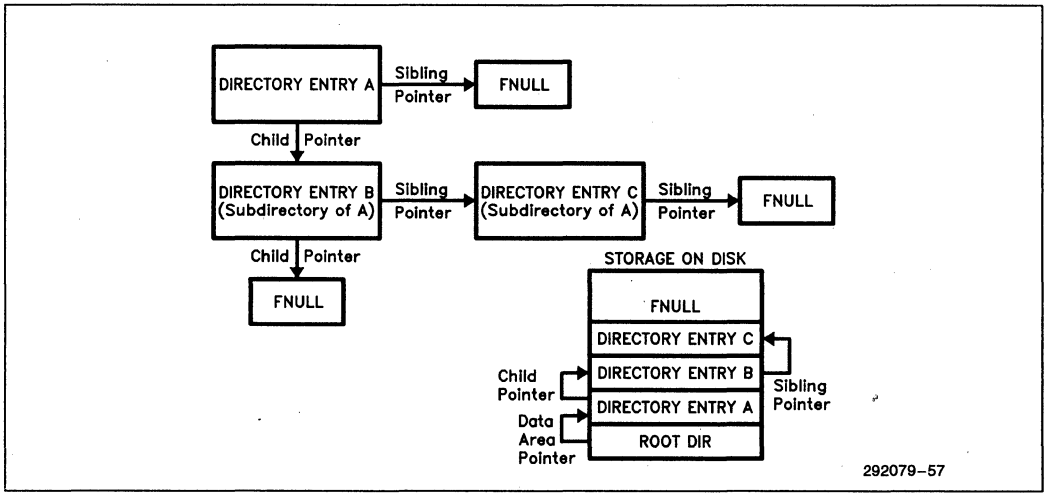


Figure 25. Directory Arrangement

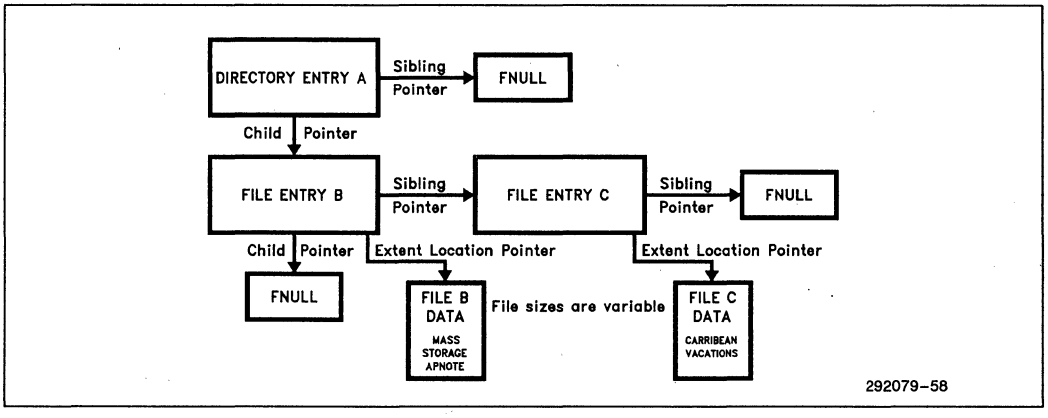


Figure 26. File Arrangement

Creating a Bootable Drive

The startup time of the PC can be decreased by booting from a flash memory disk instead of the magnetic disk. To do this, a "disk-image" is installed on the flash memory disk which is located in the system memory space between C0000H and E0000H (Figure 10). (The "disk image" contains the Boot Record, Directory, and FAT.) This memory space is referred to as Expansion ROM. During the system Power-On-Self-Test (POST), the system searches this memory area for the ROM adapter signature, 055AAh, marking the beginning of the disk image. Once this signature is found, the BOOTSTRAP process begins. The software to create and install this "disk image" is available as a product from Microsoft Corporation as ROM executable MS-DOS.

WHY FLASH?

CHARACTERISTICS OF INTEL FLASH MEMORY

Power consumption, weight, performance, and reliability are the key criteria for a system design. The discussion of Intel flash memory as a mass storage medium would not be complete without a performance analysis and comparison to other technologies.

Power Consumption

Portability of a computer demands battery longevity and consequently minimal power consumption. Small form factor disk drives are being designed specifically for the size and power requirements of laptops.

A drive has three basic operating modes: active, power savings, and standby. The active mode consists of reading, writing, and ready. Ready condition allows "instantaneous" transitions into the read or write states. In the power-savings mode only the drive motor continues to run. Standby shuts off all functionality except for the circuitry needed to "spin-up" the drive. From the standby mode, extra power and considerable time, is required to "spin-up" the disk.

Power Consumption Comparison (Watts)

(Based on typical performance characteristics. The 20 Mbyte Flash Memory disk is based on the use of 80-28F020s. Only two of the forty devices are accessed at a time, the remainder are in standby mode.)

Power Consumption (Watts)		
Active Modes	**Hard Disk Drive (2.5", 20 Mbytes)**	**INTEL Flash Memory (20 Mbytes)**
Ready	1.7–2.0	0.05 (Same as Standby)
Read	3.5-4.0	0.15
Write	3.5–4.0	0.25
Power Savings	1.5	0.05 (Same as Standby)
Standby	0.1–0.5	0.05
Spinup (from Standby)	9.3	0

For a battery-powered system, 3–4 hours of operation is unacceptable. Battery longevity is achieved by using Intel Flash Memory solid-state storage as a disk replacement. The following table relates battery life and the different functions of disk operation. A "AA" battery with a capacity of 2215 mAH is used for the comparison. Obviously, for a truly accurate representation, other components of the system should be included. But from the data storage point of view, the flash memory disk will operate many more hours than the hard disk drive on a set of batteries.

Hours of Operation for a "AA" Battery (Based on Data from Previous Table and 2215 mAH Battery Capacity)

	Hard Disk Drive (2.5", 20 Mbytes)	**INTEL Flash Memory (20 Mbytes)**
Read	0.83	22.15
Write	0.83	13.29
Standby	6.64	66.45

Data Access Time

Reading data from a magnetic disk is a very slow process compared to a solid-state disk (SSD). Disk transfer time is lengthy due to four time components: spin-up, seek time, latency, and data transfer time. Spin-up is a factor to consider for battery-powered systems, where most disk accesses are begun from the standby mode. During the seek time, the arm is repositioned to the correct track. Latency is the delay from arm repositioning until the first sector of the transfer moves under the

read/write head. This is dependent on the speed of rotation. The actual transfer of data is the third component. The standard SCSI interface transfers data between 5 Mbits and 10 Mbits per second, with which flash memory compares very favorable.

For this example it is reasonable to assume a transfer rate of 1.0 Mbytes per second. Using a full word-wide (x16) bus bandwidth (120 ns access speed of the device), flash achieves a transfer rate of 16.6 Mbytes per second.

Read Speed Comparisons

	Hard Disk (Standard SCSI Interface)	Floppy Disk	Flash Memory (16-Bit Bus, 120 ns Access)
Seek Time	28 ms		0
Latency	8.3 ms	100 ms	0
Transfer Rate	1.0 Mbyte/s	62 Kbyte/s	16.6 Mbyte/s
Total for 10 Kbyte File	46.54 ms	261.3 ms	0.62 ms

(Floppy disk drive specifications combine access into one category.)

In this example, the flash memory disk has 75 times the read performance over the hard disk. Smaller files result in even greater differences. Additionally, the 5 second spin-up of the hard disk gives the flash memory disk over 8,000 times the performance!

A byte will typically program in Intel Flash Memory in one pulse. (See Intel Flash Memory Data sheet for programming algorithm.) Based on this and the parameters used in the example above, a 10 Kbyte file is written to the flash memory disk in 87.04 ms. Because writes to a hard disk typically begin from spin-down, the flash memory disk is still over 50 times faster. Since reads are 80% of disk access, flash memory's user-perceptible performance advantage is substantial.

Reliability

The definition of hard disk mean-time-before-failure (MTBF) is extremely ambiguous. There are no industry-wide standards for making a reliable calculation. Disk drive manufacturers choose whichever method best suits their product's reliability perception.

One method uses the overall mean failure. The MTBF of all critical components is computer analyzed and the lowest one is selected. A second method tests 100 drives. The hours of the first ones to fail are multiplied by the number of drives. How many reads or writes are performed? Is the disk stopped and started during the process? Standard answers do not exist.

The vagueness of the test procedures makes it difficult to compare the MTBF for a flash memory solid-state disk and a hard disk. Based on the fact that disk usage is 80% reads and 20% writes, a reasonable comparison can be made. (What is not taken into account is that disks are an 'infinite' write, but finite read medium. Continuous reading causes reduced magnetic field strength, a failure mechanism hidden by re-writing the disk.)

Intel's Flash Memory typically performs 100,000 erase/program cycles. (Failure does not occur at this point. The only noticeable change is a gradual increase in program and erase times.) Assume a flash memory disk size of 4 Mbytes that functions like a WORM drive; it is erased and reused after filling.

Based on a typical disk MTBF of 50,000 hours and the 80/20% division, 10,000 hours are used for writing files. Assume the average file size written to disk is 10 Kbytes. A 4 Mbyte flash disk can store approximately 400 $\times$ 10 Kbyte files (4 Mbyte/10K = 408) before erasure is necessary.

These 400 files could be writen to the disk 40 $\times$ 10^6 times $-$ (400 files $\times$ 100,000 cycles = 40 $\times$ 10^6). The result is that within a 10,000 hour period, one 10 Kbyte file could be written once every 0.9 seconds.

$$\frac{10{,}000 \text{ hours}}{40 \times 10^6 \text{ Files}} \times \frac{3600 \text{ Seconds}}{1 \text{ Hour}} = \frac{0.9 \text{ Seconds}}{\text{File}}$$

It would be more realistic (although still extremely aggressive) to assume that this 10 Kbyte file is written to this disk every 10 minutes. At 100,000 cycles, 40 $\times$ 10^6 files will have been written. The MTBF can be calculated as follows:

$$40 \times 10^6 \text{ Files} \times \frac{10 \text{ Minutes}}{\text{File}} \times \frac{1 \text{ Hour}}{60 \text{ Minutes}} = 6.6 \times 10^6 \text{ Hours}$$

This is an MTBF of over 6 million hours! (See Reliability Report RR60 for more details.)

A flash memory solid-state disk outlasts its mechanical counterpart by at least two orders of magnitude, especially if head parking problems and limited start/stop cycles of the mechanical disk are taken into account.

Weight

Lowering the power consumption of your portable system also lowers the weight. Reduced battery demands mean smaller and lighter batteries and power supplies. Weight savings is also gained by the proper choice of the mass storage medium. The small 20 Mbyte 2.5″ disk drives weigh between 9 and 21 ounces. The equivalent capacity of flash memory using 80-2 Mbit TSOPs (which individually weigh 1.16×10^{-2} ounces) weighs 0.93 ounces plus the weight of the circuit board. (See section on Intel flash memory packaging.) This difference is critical when the computer weight requirement is under five (5) pounds.

SUMMARY

The advent of Intel Flash Memory has led to the evolution of solid-state mass storage. This application note has provided the building blocks that will allow the innovative manufacturer to remain on the forefront of technology.

- Advanced packaging, such as the TSOP, IC memory cards, and SIMM, is necessary for high-density applications.
- Intel Flash Memory allows flexible system interfacing by using I/O, paged, or linear addressing methods.
- Software variations enable an unlimited number of mass storage applications for Intel Flash Memory.
- Intel Flash Memory offers superior performance over the magnetic disk.

intel®

AP-361

APPLICATION NOTE

Implementing the Integrated Registers of the Series 2 Flash Memory Card

6

MARKUS LEVY
SENIOR TECHNICAL APPLICATIONS ENGINEER

October 1993

Order Number: 292096-003

Implementing the Integrated Registers of the Series 2 Flash Memory Card

CONTENTS

PAGE

INTRODUCTION 6-217

SERIES 2 COMPONENT MANAGEMENT REGISTERS 6-218

PCMCIA Release 2.0 Defined 6-218

Soft Reset Register 6-218

Global Reset-Powerdown Register 6-220

Performance Enhancement Registers .. 6-221

Sleep Control Register 6-221

CONTENTS

PAGE

Ready-Busy Status Register 6-222

Ready-Busy Mask Register 6-224

Ready-Busy Mode Register 6-225

Write Protection Register 6-228

Card Status Register 6-229

SUMMARY 6-230

GLOSSARY 6-231

RELATED DOCUMENTS 6-231

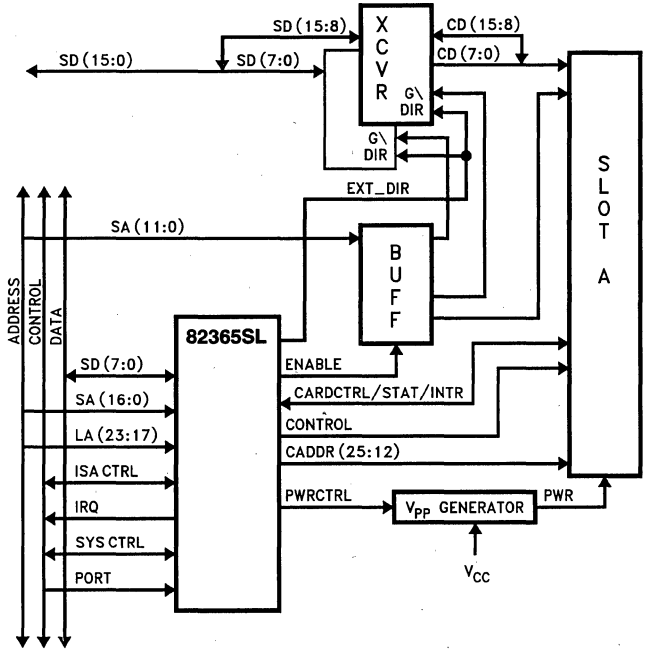
INTRODUCTION

Intel's first generation flash memory cards[1] forever changed the vision of solid-state storage. Electrically rewritable, non-volatile, reliable, yet economical in high densities, these cards provided a unique solution for the portable computing industry demanding such media. The second generation of flash memory cards provide even higher densities, lower power consumption and a higher level of functionality. The Series 2 Flash Memory Card delivers a major technology breakthrough by supporting densities up to 20 Mbytes[2], an integrated memory control register set (**Component Management Registers** or **CMRs**) and PCMCIA 2.0 compliance.

Intel's 8-Megabit FlashFile™ Memory, 28F008SA, provides the foundation for the Series 2 Flash Memory Card. Its properties include data-write and block-erase automation, sixteen 64 Kbyte, separately-erasable blocks, a Ready/Busy# output pin, and a Powerdown mode. Within the Series 2 Card, high-functionality ASICs link the flash memory devices with the PCMCIA-specified electrical interface. These ASICs handle buffering, decoding and all control signals. They also contain the CMRs and hardwired Card Information Structure (CIS) used by system software to enhance device-level functions.

The OEM has many hardware and software alternatives for using the Series 2 Card. From a hardware perspective, the Intel 82365SL offers the most practical solution for controlling the PCMCIA socket in a PC solid-state drive application. This component, called the PC Card Interface Controller, provides the ExCA compliant hardware interface between the host system and the Series 2 Cards. As shown in Figure 1, the fundamental glue logic consists of a V_{PP} generator and V_{CC} control, a latching transceiver and address and decode signal buffers. Embedded systems can provide proper card signals with discrete circuitry.

Interfacing the Series 2 Card to the 82365SL



*This solution requires minimal glue logic.

Figure 1. The 82365SL Requires Minimal Glue Logic

NOTES:
1. The Bulk-Erase iMC001FLKA, iMC002FLKA, and iMC004FLKA (One, Two and Four Megabytes, respectively).
2. Higher density cards may be realized in the future as component densities go beyond 8 Megabits.

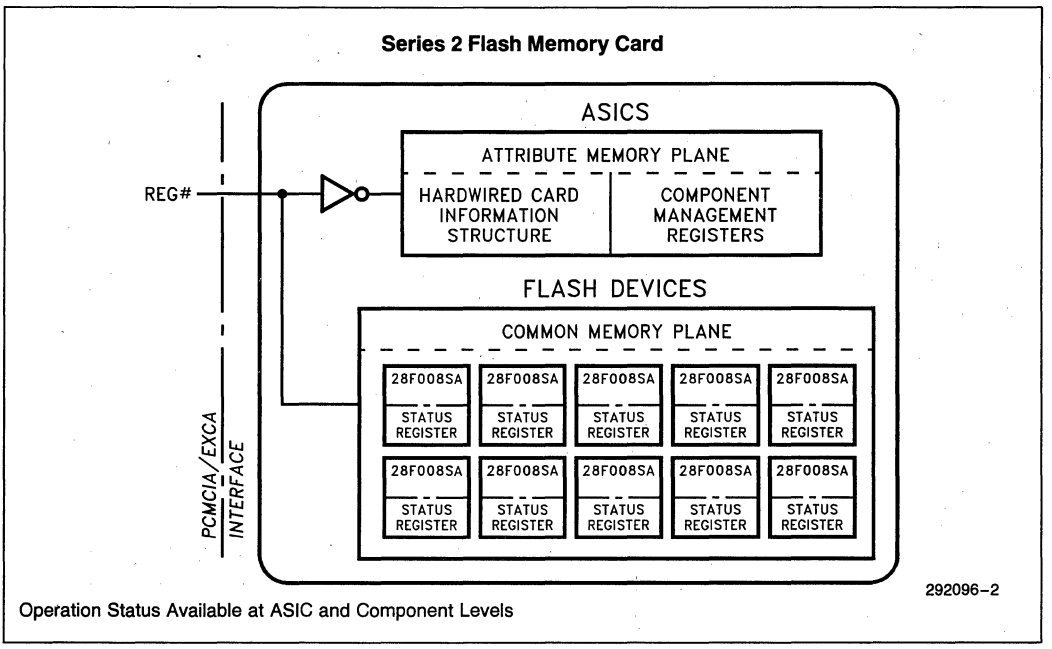


Figure 2. Selecting the Attribute or Common Memory Planes

Computer systems using the Series 2 Card as a solid-state disk drive employ file management software, such as Microsoft's* Flash File System. This software capitalizes on the architectural benefits of flash memory. It includes drivers that interface directly to the Series 2 Card. Beyond specifying the hardware architecture, PCMCIA provides a software solution that consists of modular software pieces designed for easy adaptation to the various hardware platforms and memory technologies. The various pieces of the PCMCIA system may be obtained from your BIOS vendor. Essentially, this means that a system OEM is relieved of having to write any software for Series 2 Flash Memory Card.

This application note supplements the information contained in the Series 2 Flash Memory Card Data Sheet. It benefits OEMs developing their own Series 2 Flash Memory Card software pieces, including custom flash file management software and software for embedded systems running non-DOS applications. Specifically, it describes the software aspects of implementing the card's CMRs which provide software control of many 28F008SA functions, elevating the system designer above device-level issues used by higher-level file system software.

SERIES 2 COMPONENT MANAGEMENT REGISTERS

The CMRs optimize the Series 2 Flash Memory Card's performance by supplying a software-controlled interface to the individual devices within the card. As shown in Figure 2, they are accessed as memory-mapped I/O in the Attribute Memory Plane by pulling the card's Register Select pin low (REG#, pin 61)(3). CMRs can be divided into two basic categories; those defined by the PCMCIA Release 2.0 specification and Intel defined "Performance Enhancement Registers".

PCMCIA RELEASE 2.0 DEFINED

Soft Reset Register
(Configuration Option Register)

During card operation, it may be necessary to place the card into a known state by resetting the 28F008SA-level Status Registers and the CMRs in the ASICs to their power-on conditions (Figure 3). Specifically, in the

NOTE:
3. No switch-over setup-time from Common Memory is needed when PCMCIA timing requirements are met.

Component Management Registers[4]

Defined by the PCMCIA R2.0 specification
- **Soft Reset Register [5]—(R/W)**
- **Global Reset-Powerdown Register [6]—(R/W)**

PERFORMANCE ENHANCEMENT REGISTERS designed to deliver control benefits tied directly to the Intel 28F008SA:

- **Sleep Control Registers—(R/W)**
- **Ready-Busy Status Registers—(RO)**
- **Ready-Busy Mode Registers—(R/W)**
- **Ready-Busy Mask Registers—(R/W)**
- **Write Protection Registers—(R/W)**
- **Card Status Register—(RO)**

ASICs, this reset affects the RP bit (**Global Reset-Powerdown Register**), the **Sleep Control Register**, the **Ready-Busy Mode Register**, the **Ready-Busy Mask Register**, and the CISWP and CMWP bits (**Write Protection Register**). There are several ways to enter power-on status:

1. Issuing a hardware reset, with a complete system reset or socket reset through the interface hardware, affects the entire system or the Series 2 Card, respectively.
2. During normal operation of many portable systems, tremendous power savings are realized by entering a suspend state. In this state, power to the card's socket is removed. After reapplying power, the card automatically attains its power-on status. Therefore, before removing power from the Series 2 Card, system software must save the contents of the Component Management Registers. It should also be pointed out, that a startup period must elapse to allow all internal circuitry to stabilize before accessing the card. This time period depends on host system power supply capabilities.[7]
3. The third method utilizes a software-controlled mechanism built into the Series 2 Card. This option, activated with the **Soft Reset Register**, provides a simple approach for placing the card in its power-on state without time delay.

The **Soft Reset Register** (Figure 4) contains a soft reset (SRESET) bit that performs a function similar to the hardware reset invoked by the card's RESET pin (RST#, pin 58)[8]. Achieve the reset condition by issuing a two-step write sequence to the SRESET bit (i.e. toggling from 0 to 1 and back to 0).

During reset (SRESET = 1), the ASICs drive the flash memory array into the deep-sleep mode. This aborts any device operations in progress and resets each device's Status Register. After initiating a soft reset, the SRESET bit *must* be cleared (zero) to enable access to the flash memory array or write to another CMR. The host system can clear this bit by writing in a zero or issuing a hardware reset.

6

Power-On Conditions*

ALL DEVICES IN STANDBY MODE.
SOFTWARE WRITE-PROTECT DISABLED.
ALL DEVICES' READY/BUSY# OUTPUTS UNMASKED.
PCMCIA-READY/BUSY# MODE ENABLED.
READY/BUSY# OUTPUT PIN GOES TO READY.

NOTE:
Generated by Hardware Reset or Toggling SRESET Bit.

Figure 3

NOTES:
4. R = READ, W = WRITE, RO = READ ONLY
5. Referred to as Configuration Option Register by PCMCIA R2.0.
6. Referred to as Configuration and Status Register by PCMCIA R2.0.
7. As specified by PCMCIA Release 2.0.
8. Soft reset puts all devices into power-down mode and requires a recovery time after returning from soft reset (500 ns for reads and 1 μs for writes).

Soft Reset Register
(Configuration Option Register)

PCMCIA-Defined

CIS ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4000H	SRESET	PCMCIA CONFIGURATION INDEX POWERS UP AND RESETS TO ZERO						

- Toggle SRESET to reset and return to standby-mode.
- For power-on default, SRESET = 0.

Figure 4. Useful for placing the card into a known state

The other two fields (not implemented with the Series 2 Card), defined in this register by the PCMCIA R2.0 specification, include the Configuration Index and the LevlREQ. After powerup or soft reset, the Configuration Index contains zeros to maintain compatibility as a Memory-Only Interface. The LevlREQ bit is hardwired to zero.

GLOBAL RESET-POWERDOWN REGISTER

PCMCIA R2.0 Defined

(Configuration and Status Register)

The portable system designer strives to minimize power consumption in every conceivable way. Solid-state storage devices using Intel Flash Memory deliver significant power consumption reductions (when compared to the mechanical disk) and therefore play an important part of the system design considerations. The portable system aimed at power conservation looks to shut down portions of system circuitry not in use (i.e. the solid-state drive not accessing files, the screen's backlight when the keyboard has not been touched in a certain amount of time, etc.). Powering down the entire socket achieves a minimal power usage status. However, the powerup recovery time from this approach produces varying delays.

The Series 2 Card offers the optimal solution with the **Global Reset-Powerdown Register** (Figure 5). Writing a one (1) to the Reset-PwrDwn Bit (RP bit 2) of this register puts all internal devices into the Deep-Sleep Mode by pulling every device's RP# input low(9). In the Deep-Sleep mode, a 20 Megabyte Series 2 Card consumes 90% less current versus the standby mode current(10).

When the host system drives the two card enable pins high(11), the Series 2 ASIC circuitry blocks system-level address and data signals from the internal devices. Additionally, latching address buffers and data transceiv-

Global Powerdown Register
(Configuration Option Register)

PCMCIA-Defined

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4002H	ZEROS					RP	ZEROS	

- Reset-Powerdown places all devices into Deep-Sleep mode.
- Write zeros to maintain PCMCIA compatibility.
- RP = 0 after reset.

Figure 5

NOTES:
9. The remaining fields in this register (Changed, SigChg, IOis8, Audio, Intr and Rsvd) are tied low in the Series 2 Card for PCMCIA compatibility and for simplifying software masking.
10. I_{CCS} = 30 μA vs, I_{CCSL} = 0.2 μA; refer to 28F008SA Data Sheet. The ASICs consume 1 μA.
11. CE1# (pin 7) and CE2# (pin 42) = V_{IH}

SAMPLE 80X86 CODE TO HANDLE RECOVERY-PERIOD TIMING

```
   GLOBAL_RP      EQU      4002H       ;Global Reset-PowerDown Register
   NOT_RP      EQU      0H

   MOV AX, MEM_CARD_BASE       ;Load card address
   MOV ES, AX
   MOV DI, GLOBAL_RP           ;Pointer setup

;Software assumes already in REG# mode access
   MOV BYTE PTR ES:[DI], NOT_RP ;Clears RP bit

   MOV CX, RECOVERY_TIME       ;Based on speed of processor
 FOR_A_WHILE:
   LOOP FOR_A_WHILE
```

Figure 6. Assembly Language Code for Returning from "Deep-Sleep" Mode

ers on the host side eliminate address and data signal switching at the Series 2 Card input buffers further reducing power consumption levels. In other words, to achieve the lowest power consumption levels, these signals should not be floated or tristated.

After clearing the RP bit, the device-recovery times must be met before accessing the flash memory. As shown in Figure 6, the recovery period can be implemented using a simple software algorithm(12).

Prior to entering the Reset-Powerdown Mode, your software must check operation status for data-writes or block-erases in progress(13). Any operations in progress will be terminated when powering down the flash array. The 28F008SA does not maintain Status Register contents in the Reset-Powerdown Mode. Therefore, when the card returns to standby mode, all devices will report successful status (Status Register = 80H) indicating the need for software drivers to use the reset-powerdown function intelligently.

PERFORMANCE ENCHANCEMENT REGISTERS

Sleep Control Register
(Performance Enhancement Register)

The reset-powerdown functionality of the **Global Reset-Powerdown Register** has a global affect on all devices. In many solid-state storage applications, reading or writing files only requires access to select device pairs and the remaining devices could be kept in Deep-Sleep status until needed.

Sleep Control Registers
Performance Enhancement Register

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
411AH	RESERVED						DEVICES 18/19	DEVICES 16/17
4118H	DEVICES 14/15	DEVICES 12/13	DEVICES 10/11	DEVICES 8/9	DEVICES 6/7	DEVICES 4/5	DEVICES 2/3	DEVICES 0/1

- For reset, all devices powered up (bits = 0).
- On cards less than 20 megabytes, absent devices read as "0"s.
- Bits cleared to zero by SRESET and RESET.

Figure 7. Allows Selective Reset-Powerdown of Devices within the Series 2 Card

NOTE:
12. PCMCIA does not specify a maximum recovery time. Recovery times, varying for different card technologies, must be handled on a case-by-case basis.
13. Polling the individual device's Status Register, the Ready/Busy Status Register, or the RDY/BSY# bit in the Card Status Register.

28F008SA Status Register Bit Definition

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
WSM STATUS	ERASE SUSPEND STATUS	ERASE STATUS	WRITE STATUS	VPP STATUS	RESERVED		

Figure 8. Read during Write or Erase Operations to Determine Status

The **Sleep Control Register** (Figure 7) offers this option; each bit provides power down for a specific device pair. Except for the global vs individual affect, this register functions identically to the **Global Reset-Powerdown Register**. The global reset-powerdown can be enabled while individual devices are sleeping. Disabling the global RESET-PWRDWN does not affect prior bit settings of the **Sleep Control Register**.

In many applications using the Series 2 Card, the card will be in the Standby Mode a large percentage of the time. This avoids device recovery times associated with complete socket power off or entering the Deep-Sleep Mode. In the Standby Mode, the **Sleep Control Register** offers the greatest advantage over the **Global Reset-Powerdown Register**. With the capability of controlling individual device pairs, a power savings improvement of approximately 16 times (based on typical current values) will be seen. This is derived from the following information:

- 28F008SA devices in Deep-Sleep; I_{CC} = 0.2 μA, I_{PP} = 0.1 μA.
- 28F008SA devices in Standby; I_{CC} = 30 μA, I_{PP} = 1 μA.
- ASICs in Standby and Sleep; I_{CC} = 1 μA.
- With device-pair control, unaccessed devices remain in Deep Sleep.

Although the other operating modes (read, data-write, or block-erase) also experience power savings by using the **Sleep Control Register**, the effects are not as significant relative to the higher current requirements of those modes.

When using the **Sleep Control Register**, software must account for the same device-recovery time of the global reset-powerdown method. To access files (or data) that span multiple device pairs (and experience uninterrupted access), software can perform a "look-ahead" function to determine which device pairs must be powered up.

READY-BUSY STATUS REGISTER

Performance Enhancement Register

The automated data-write and block-erase capability of the Intel 28F008SA FlashFile Memory results in a significant performance improvement. Furthermore, automation simplifies system-level interfacing as the user only delivers the proper command and monitors the operation's READY/BUSY status. Referring to the 28F008SA Data Sheet (or Figure 8), operation status can be obtained from the device's *Status Register* or RY/BY# pin. The device's Status Register allows software polling for ready status in addition to write and erase status. The RY/BY# pin can be used to generate an interrupt when making a busy to ready transition. Regardless of the method used for determining ready status, the Status Register should be read to determine whether an operation was successful.

In the Series 2 Card, where multiple devices are present and multiple simultaneous operations can occur, software polling each device's Status Register requires extra software and time. Furthermore, the PCMCIA interface only has one RDY/BSY# pin which obviously prevents 20 devices from hooking their individual RY/BY# out to the system. The ASICs within the card take these signals and feed them into the **BUSY# Status Register** (Figure 9). This facilitates multiple device-pair operations by allowing an analysis of all devices simultaneously. After initiating the data-write and block-erase operations, the system can switch the card to the Attribute Memory Plane to access these registers. Alternatively, each device's RY/BY# signal funnels into a single "wired or" signal that becomes the PCMCIA-RDY/BSY# pin driving an interrupt or polled through an I/O port.

When performing single device pair operations, Ready/Busy status should be accessed directly from the Status Register of the flash memory devices for the following reasons: 1) A device's Status Register must be read anyway to determine the result of an operation; 2) This saves several instructions required to switch to the Attribute Memory Plane.

Ready-Busy Status Register
Performance Enhancement Register

CIS ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4134H	RESERVED				DEVICE 19	DEVICE 18	DEVICE 17	DEVICE 16
4132H	DEVICE 15	DEVICE 14	DEVICE 13	DEVICE 12	DEVICE 11	DEVICE 10	DEVICE 9	DEVICE 8
4130H	DEVICE 7	DEVICE 6	DEVICE 5	DEVICE 4	DEVICE 3	DEVICE 2	DEVICE 1	DEVICE 0

- Each bit corresponds to a device's RY/BY# signal.
- Devices not present (i.e. < 20 Megabytes) return ready status.

Figure 9. Monitors Individual Device's RY/BY# Pins

Example for Monitoring Ready/Busy Status

(Assume ES contains memory card base address)

```
RDY_BSY_STATUS          EQU 4130H ;Register address
DEVICE_0                EQU 01H   ;Settings in register for specific devices
DEVICE_1                EQU 02H
DEVICE_2                EQU 04H
DEVICE_3                EQU 08H
DEVICE_4                EQU 10H
DEVICE_5                EQU 20H

XOR AX, AX                              ;Zero AX Register
MOV DI, RDY_BSY_STATUS

;Insert code to start write operation in first 3 Device Pairs
;i.e. Devices 0, 1, 2, 3, 4, 5.

OR AX, DEVICE_0
OR AX, DEVICE_1
OR AX, DEVICE_2
OR AX, DEVICE_3
OR AX, DEVICE_4
OR AX, DEVICE_5

;Assume card already in REG mode.
TEST BYTE PTR ES:[DI], AX       ;Zero flag cleared when programming
                                ;devices are ready.
```

6

READY-BUSY MASK REGISTER
Performance Enhancement Register

As described earlier, completion of a data-write or block-erase operation can be determined by attaching the card's RDY/BSY# pin into a system interrupt. This frees the host system to perform alternate tasks after initiating an operation. In other words, device-level automation allows Series 2 Card operations to become background tasks.

Occasions exist where the interrupt generated from a device becoming ready produces unacceptable latency times. For instance, data-write operations, completing in only 10 μs, realize a performance penalty dealing with interrupt latencies longer than the write time itself. The data-write operations would achieve a higher level of performance by using software polling techniques[14]. On the other hand, block-erase operations typically require one second. Therefore, these operations perform well as background tasks because the interrupt latency constitutes a small fraction of the total time.

This discussion implies that the system interrupt should be disabled for data-writes and enabled for block-erases. What if an application requires simultaneous writes and erases? The Series 2 Flash Memory Card handles this situation with its Ready-Busy Mask Register (Figure 10). Setting the appropriate mask bits in the Ready-Busy Mask Register blocks the corresponding device's RY/BY# signals. With a device's mask bit set, the card's RDY/BSY# pin and Card Status Register (bit 0) always reflect a ready condition, regardless of the operation status. Figure 11 displays a conceptual mask circuit for a single device. The mask settings have no effect on the card's Ready-Busy-Status Registers (providing direct access to each device's RY/BY# output) or the Device Status Register. This allows software polling in the usual manner.

Ready-Busy Mask Register
Performance Enhancement Register

CIS ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4124H	RESERVED				DEVICE 19	DEVICE 18	DEVICE 17	DEVICE 16
4122H	DEVICE 15	DEVICE 14	DEVICE 13	DEVICE 12	DEVICE 11	DEVICE 10	DEVICE 9	DEVICE 8
4120H	DEVICE 7	DEVICE 6	DEVICE 5	DEVICE 4	DEVICE 3	DEVICE 2	DEVICE 1	DEVICE 0

1 = MASKED

Figure 10. Allows Masking of Individual Device's Ready/Busy Signals

Selecting the Appropriate Device to Mask

Assume the register set DI:DX contains a 32-bit physical address into SERIES 2 card.
Each device pair represents 2 Megabytes (i.e. 200000H).

```
MOV CL, 5  ;Load shift count
SHR DI, CL ;Result in DI is device pair number to mask.

;Now determine whether to mask device pair for word operations or use Bit 0 of
the DX portion to determine high or low device (odd or even) for byte
operations.
```

NOTE:
14. Polling the individual device's Status Register, the Ready/Busy Status Register, or the RDY/BSY# bit in the Card Status Register.

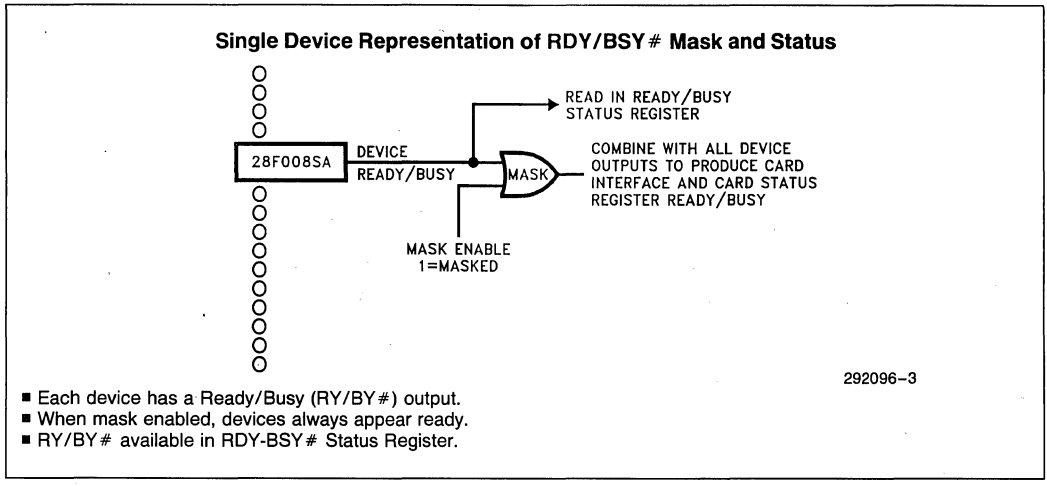


Figure 11. The Ready-Busy Mask is Very Useful for Write Optimization

READY-BUSY MODE REGISTER

Performance Enhancement Register

The PCMCIA specification for the Ready/Busy interface states that "*the RDY/BSY# line is driven low by the memory card to indicate that the memory-card circuits are busy and unable to accept a data-transfer operation.*" Contrary to the PCMCIA specification, device-level data-write and block-erase automation enables the Series 2 Card to perform multiple operations simultaneously. Using the PCMCIA-specified method of RDY/BSY# functionality for multiple device operations, the RDY/BSY# interrupt does not notify the system until all devices finish because busy devices hold the RDY/BSY# signal low, as shown in Figure 12. Multiple block erases (typical block erase time of 1 second) could present an unacceptable pushout if system software waits for the first available "clean" block to write data.

The Series 2 Card offers an alternative Ready/Busy mode (High-Performance Ready/Busy mode, alias "Levy"-mode) removing the performance impact of the PCMCIA mode. Circuitry internal to the ASIC catches every "READY-going" edge from each device. After an individual device becomes ready (Ready/Busy signal goes high), the system has the opportunity to immediately service the interrupt. System software must now toggle the CLEAR bit (bit 1) in the Ready-Busy Mode Register (Figure 14) to reactivate the Ready/Busy signal. Figure 13 demonstrates the resulting waveform.

The Series 2 Card powers up in the PCMCIA-mode. Switching into the High Performance mode requires a two step process, as shown in Figure 15. ASIC circuitry design prevents being able to write a zero to the RACK bit on the same cycle as entering the High-Performance RDY/BSY Mode. This intentional design technique eliminates the possibility of receiving a noise generated RDY/BSY# rising edge, which would trigger an unwanted interrupt.

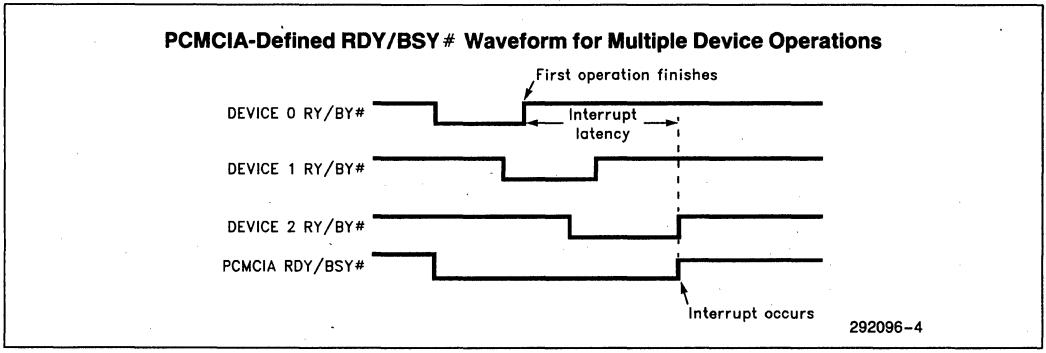
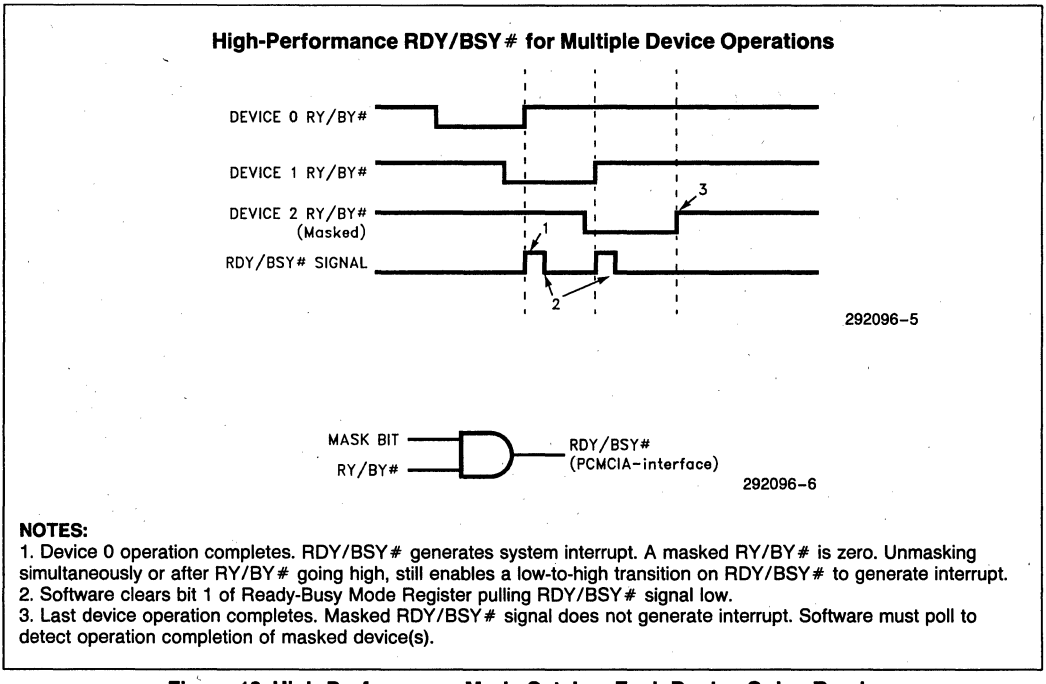


Figure 12. PCMCIA-Defined RDY/BSY Waveform for Multiple-Device Operations



NOTES:
1. Device 0 operation completes. RDY/BSY# generates system interrupt. A masked RY/BY# is zero. Unmasking simultaneously or after RY/BY# going high, still enables a low-to-high transition on RDY/BSY# to generate interrupt.
2. Software clears bit 1 of Ready-Busy Mode Register pulling RDY/BSY# signal low.
3. Last device operation completes. Masked RDY/BSY# signal does not generate interrupt. Software must poll to detect operation completion of masked device(s).

Figure 13. High-Performance Mode Catches Each Device Going Ready

Ready/Busy Mode Register
Performance Enhancement Register

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4140H	RESERVED						RACK	MODE

- Mode = Ready/Busy Mode
  0 = PCMCIA Mode
  1 = High-Performance Mode
- RACK = Ready Acknowledge Bit
  Clear this bit after receiving ready status to prepare for next device's ready transition
- Register defaults to PCMCIA Mode for power on or reset. In PCMCIA Mode, RACK is a Don't Care

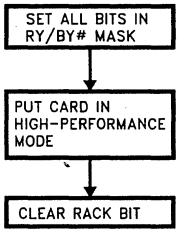
Figure 14. To Prevent Accidental Ready Transitions, a Three Step Sequence must be Followed to Enter High-Performance Mode

As discussed in the previous section, the block-erase operation benefits from the interrupt capabilities of the RDY/BSY# signal. However, if your software only erases one device pair at any time, the PCMCIA-RDY/BSY# Mode will be sufficient for two reasons: 1) Both devices started simultaneously will complete the erase operation almost at the same time; 2) in 16-bit access mode, both devices of the pair must be erased before writing.

To block-erase in multiple devices:

1. Be sure to mask all devices (in Ready/Busy Mask Register).
2. If not already done, place the Series 2 Card in the High-Performance Mode (refer to Figure 15).
3. Issue the block-erase command sequence to the appropriate devices.

Enabling High-Performance Ready/Busy Mode



Prevents ready devices from triggering an unwanted rising edge, and generating an interrupt after clearing RACK bit.

Write a one (1) to the Mode bit of the RY/BY# Mode Register.

Write a zero (0) to the RACK bit of the RY/BY# Mode Register. The hardware requires this sequence to eliminate unwanted interrupts caused by signal-bounce.

Figure 15. Entering High-Performance Mode

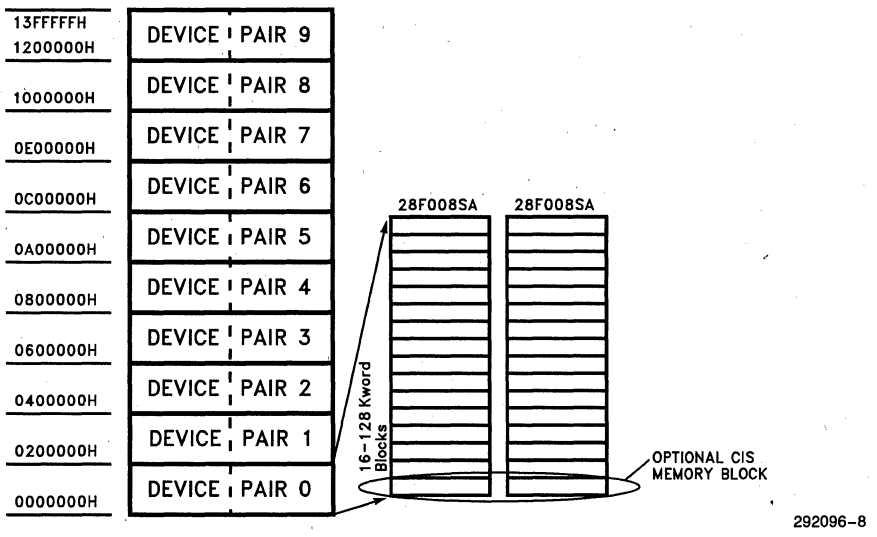
4. Unmask appropriate **Ready-Busy Mask Register** bits. The circuitry catches devices with already completed erase operations with the conceptual setup shown in Figure 13. Use a RAM-based variable or register for an erase-block queue to monitor erasing devices.

The interrupt service routine (ISR) can be as simple as removing the erase block from the queue. It could also be used to notify the system that this block is free to use. Regardless of the ISR implementation, it should include the following basic procedures:

1. Set all RY/BY# masks in the **Ready-Busy Mask Register**. This prevents additional interrupts within the ISR (i.e. prevent re-entrant interrupt). Keep track of mask setup to reinstate before ISR exit.
2. Check the queue of erasing devices and read the Ready-Busy Status Register to determine which device completed the operation.
3. Service the erased block(s). Even though one erased block generated the interrupt, more blocks may have completed erasing at this point.
4. Clear RACK in the Ready-Busy Mode Register.
5. Before exiting the ISR, reset the mask. This "catches" devices that went ready during the ISR and will cause a re-entrant ISR. However, at this point in the ISR, this will not affect system or software integrity.

WRITE-PROTECTION REGISTER

The Series 2 Card contains a PCMCIA-defined, hardwired Card Information Structure (CIS) accessed in the Attribute Memory Plane. This data structure provides fundamental, unchanging information pertaining to the card. It includes card size, type of components, access speed, etc. Situations exist where the user needs to include custom-format information, such as card partitioning and operating system specific information.



- The first block pair of the first device pair is the common memory CIS; write protect using CISWP bit of Write Protection Register.
- Write protect the remaining 159 block pairs using the CMWP bit of the Write Protection Register.

Figure 16. The WRITE PROTECT REGISTER Blocks Writes to the Two Sections of the Common Memory Plane

This information can be loaded in the Common Memory CIS during card format (refer to Figure 16). Typically, after writing this information, it would rarely change. The Series 2 Card provides a means of locking this area of memory, as well as the remainder of the Common Memory array with the **Write Protection Register** (Figure 17). The **Write Protection Register** has an advantage over the mechanical write protect switch in that it allows software to control user write access to the card's data (the mechanical switch can be easily switched off enabling card writes). For example, a pen-based system may use this feature to protect its read-only operating system stored within the Series 2 Card.

The CIS Write Protect Bit (CISWP, bit 0) prevents writes to the Common Memory CIS blocks. When software determines that this block of memory contains valid, *custom-format* information (contains PCMCIA tuple data structure), the CISWP Bit could be set to prevent accidental data corruption by another application. Note that if an End-User format utility is provided, this software must be careful not to destroy the custom format information which could be accessed if the CISWP Bit was deactivated. The Common Memory Write Protect Bit (CMWP, bit 1) prevents writes to the remainder of the Common Memory Plane (i.e. minus the Common Memory CIS blocks). To "software"-write-protect the *entire* Common Memory Plane, both bits must be set.

Write Protection Register
Performance Enhancement Register

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4104H	RESERVED						CMWP	CISWP

- CISWP = Common Memory CIS Write Protect
- CMWP = Common Memory Write Protect
- "1" = Write Protected

Figure 17. Provides a Software Implementation of the Write Protect Switch

CARD STATUS REGISTER

Performance Enhancement Register

This (Read-Only) register provides quick access to generalized conditions within the Series 2 Card (Figure 18). It provides a shorthand method for checking the following functions:

- Ready/Busy Status
- Ready/Busy Masking
- Deep-Sleep Modes
- Setting of Mechanical Write-Protect Switch
- Software Write Protect Status
- Soft Reset Status

Where the RY/BY# Bit (bit 0, **Card Status Register**) displays the operation status of the cumulative devices within the card, the **Ready-Busy Status Registers** reflects the status of each individual device. Bit 0 (RDY/BSY#) mirrors the card's RDY/BSY# (Ready/Busy) output pin, also reflecting any Ready/Busy masking conditions. Two circumstances warrant the use of this bit: 1) When the hardware interrupt triggered by the RDY/BSY# pin produces an unacceptably long latency period, this bit should be software polled instead to increase performance; 2) When multiple devices have data-write/block-erase operations in progress, reading this cumulative Ready/Busy status will be quicker than reading multiple status registers within each device. However, when the application requires immediate access to each device as it finishes an operation, individual Device Status Registers or the card's **Ready-Busy Status Register** must be used.

Card Status Register
Performance Enhancement Register

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
4100H	ADM	ADS	SRESET	CMWP	PWRDWN	CISWP	WP	RDY/BSY#

- RDY/BSY# = Reflects PCMCIA interface RDY/BSY# pin, 0 = busy
- WP = Mechanical Write Protect Switch, 0 = off
- CISWP = Common Memory CIS Write Protect, 0 = off
- RP = Reset-Powerdown Reflects RP in Global Reset-Power Down Reg, 1 = Power Off
- CMWP = Common Mode Write Protect, 0 = Off
- SRESET = Soft Reset Reflects SRESET in SOFT RESET Reg, 1 = Soft Reset
- ADS = Any Device Pair Powered Down, OR'd Condition of Sleep Control Reg, 1 = Power Off
- ADM = Any Device Masked, OR'd Condition of Ready-Busy Mask Reg, 1 = Masked

Figure 18. Provides Generalized Card and Device Information

Bit 1 reflects the card's mechanical switch position (1 = Write Protected). This switch disables any writes to the card. Two software strategies can be implemented for this bit: 1) Assume the card's Write-Protect switch is off. Attempt to write to the card and only check the Write-Protect status if the data-write fails (which it will if the switch is on); 2) Check the switch first to avoid the possibility of failing a data-write. The choice depends on the application.

Bits 2 (CISWP = Common Memory CIS) and 4 (CMWP = Common Memory Write Protect) are direct (Read Only) inputs from the **Write-Protect Register**. These bits should be checked in a manner similar to that for Bit 1 (WP). For more detail refer to the **Write-Protection Register** section.

The RP Bit (bit 3) provides a (Read Only) version of the RP Bit in the **Global Reset-Powerdown Register** (1 = RP). Only the Attribute Memory Plane is available with the Reset-Powerdown feature enabled, allowing access to the **Component Management Registers**.

The SRESET Bit (bit 5) provides a (Read-Only) version of the SRESET Bit in the **Soft Reset Register** (1 = Locked in soft reset state).

SUMMARY

The Series 2 Flash Memory Card delivers the hardware capabilities required for implementing a solid-state storage device. Software engineers will find the features of this card both flexible and powerful when coupled with flash-optimized filing systems, such as Flash File System from Microsoft. This application note has discussed the various methods of using the **Component Management Registers** to facilitate designs incorporating the SERIES 2 card.

- PCMCIA-Defined Registers provide generalized assistance for memory card interfacing.
- Performance Enhancement Registers boost software control over the card's internal flash memory devices.

GLOSSARY

Attribute Plane: Memory plane within the card selected by pulling the REG# pin low. This random access memory contains the CIS and Component Management Registers.

Block-Erase: Erasing sections of a single flash memory device.

Bulk-Erase: Erasing the entire flash device simultaneously.

Common Memory: The memory card's main memory array.

Common Memory-Card Information Structure: The first block pair of the first device pair. Useful for storing custom format information, such as partitioning of the card.

Component Management Registers (CMR): Memory-mapped I/O registers used to control device-level functions.

Deep-Sleep Mode: A special very low power mode useful for saving power when not accessing the flash memory components.

Device-Pair: Arrangement of the 8-bit 28F008SA devices in the SERIES 2 card in a word-wide manner.

Hardwired Card Information Structure (CIS): Embedded into the Attribute Memory Plane to describe non-changing information about the SERIES 2 Card (i.e. density, speed).

Personal Computer Memory Card International Association (PCMCIA): The organization formed to promote interchangeability of IC cards by providing a standardized mechanical, electrical and metaformat interface.

Performance Enhancement Registers: Memory-Mapped I/O registers included by Intel in the Series 2 Card to boost performance by providing software control of the internal 28F008SA functions.

Ready/Busy: Indicator used to determine when a data-write or block-erase operation has completed. Symbolized by RY/BY# for the 28F008SA and RSY/BSY# at the Series 2 Card interface.

Status Register: A register internal to a 28F008SA FlashFile™ Memory device used to determine write and erase operation status.

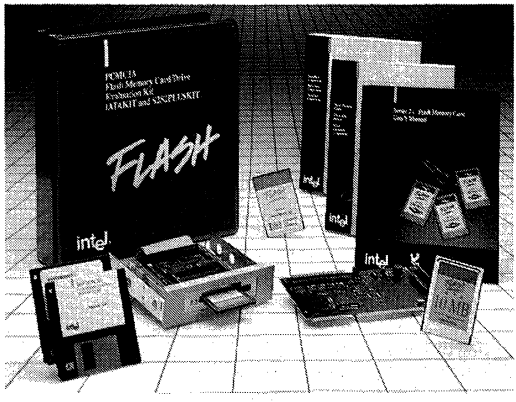
RELATED DOCUMENTS

28F008SA, 8 Megabit, FlashFile™ Memory Data Sheet
Series 2 Flash Memory Card Data Sheet
82365SL, PC Card Interface Controller Data Sheet
PCMCIA PC Card Standard Release 2.0

Flash Memory Card and Flash Drive Evaluation Kit Product Brief

Product Highlights

- **Complete kit for adding a PCMCIA socket to an AT-based system**
- **The Flash Drive Kit (iATAKIT) includes a 5MB Flash Drive**
- **The Flash Card Kit (S2S2PLUSKIT) includes both a 4MB Series 2+ and a 2MB Series 2 Flash Card**
- **Includes ISA-based PCMCIA Reader/Writer that fits into an empty floppy disk slot on your system**
- **Created for x86 DOS- or Windows*-based systems**
- **Includes source code for non-DOS system development**
- **Includes SystemSoft's* CardSoft* suite of software**
- **Contains industry-standard PCMCIA software, including SystemSoft Flash File System (FFS) software**
- **Contains a 68-pin PCMCIA-ATA to 40/44-pin IDE adapter for easy conversion (iATAKIT only)**
- **Excellent for evaluating Flash Card or Flash Drive performance with multiple software configurations**



Product Description

The Flash Drive (iATAKIT) and the Flash Card (S2S2PLUSKIT) evaluation kits are designed to make DOS-based AT systems Intel Flash Card or Flash Drive-ready — quickly and simply. Each kit provides everything necessary to add a PCMCIA socket to a desktop system.

The Flash Drive and Flash Card kits are ideally suited for designers creating PDA-like products; designers of mobile, handheld and sub-notebook products; designers who want to evaluate Flash Card and/or Flash Drive products for embedded applications; and ISVs who write software for Flash Card or Flash Drive products.

To shorten time to market, both kits contain all necessary tools in a single box. Thus eliminating the time-consuming task of seeking and obtaining separate components.

Each kit includes SystemSoft's CardSoft suite utilizing industry-standard PCMCIA software — as well as SystemSoft's Flash File System (FFS) software. Additionally, each kit includes low-level driver and MTD source code, enabling modification for your specific requirements or product differentiation. For easy conversion of PCMCIA-ATA to IDE pinouts a 68-pin to 40/44-pin adapter is included.

intel®

Features	Benefits
— All-in-one kit	— Contains everything needed to develop Intel Flash Card or Flash Drive-ready systems
— Includes low-level driver and MTD source code	— Enables code modification for specific requirements
— SystemSoft CardSoft	— Industry-standard PCMCIA software suite
— Flash File System (FFS) software	— Industry-standard file system from SystemSoft
— 68-pin ATA to 40/44-pin IDE adapter	— Easily converts PCMCIA-ATA pinout to IDE

The Intel logo is a registered trademark of Intel Corporation.
Other brands and names are the property of their respective owners.

Printed in USA/0694/40K/ASI/LK
©1994 Intel Corporation

Order Number: 297506-001

intel®

Intel FlashFile™ Memory The Key to Diskless Mobile PCs

JANET WOODWORTH
MEMORY COMPONENTS DIVISION

March 1994

Order Number: 297115-001

INTRODUCTION

As the PC evolves into what is truly a "personal" computer—one that can be held in your hand—a completely different system memory architecture will emerge. Step aside ROM, DRAM, floppy disk, and hard disk; Intel's FlashFile™ memory is here. FlashFile memory will finally make it possible to build a thin, 2-pound notebook computer that runs for many hours on a few AA batteries. But before these mobile PCs are built, designers must learn some new ways to configure system memory.

In April 1992, Intel introduced a new flash memory architecture with a combination of functionality and price that redefines mobile computing. This new architecture, when implemented in new system memory configurations, enables nonvolatile executable system memory and removable file and program storage. Intel's new flash architecture lets designers create a portfolio of products that will clearly differentiate them from their competition.

WHY A NEW MEMORY ARCHITECTURE?

The ideal memory system is:

- Dense (stores lots of code and data in a small amount of space and weighs very little)
- Fast (lets you read and write data quickly)
- Inexpensive (low cost per megabyte)
- Nonvolatile (data remains when power is removed)
- Power Conscious (prolongs battery life and reduces heat)
- Reliable (retains data when exposed to extreme temperature and mechanical shock)

Since PCs were introduced over 10 years ago, designers have grappled with how to construct memory systems that offer all these attributes. They have wisely elected to use to optimum combination of solid-state memory and magnetic storage, i.e., DRAMs plus magnetic hard disks. DRAMs are dense and inexpensive, yet slower than the processors they serve, and they are volatile. SRAMs are used in caching schemes to compensate for DRAM's slowness. While SRAMs keep pace with today's high-performance microprocessors, they are not as dense as DRAM, are inherently more expensive, and volatile. Magnetic hard disks are very dense, inexpensive on a cost-per-megabyte basis, and nonvolatile, but they are painfully slow, power hungry and subject to damage from physical shock.

ENTER FLASH MEMORY

Because Intel's ETOX™ III flash memory cell is 30-percent smaller than equivalent DRAM cells, the company expects it to track DRAM density closely. Intel's new 28F008SA FlashFile Memory can store 8 megabits, or one megabyte, of data today. Flash memory is more scalable than DRAM due to its simple cell structure, so as DRAM technology shrinks towards 0.25 microns and 64 megabits, flash will pace and ultimately overtake DRAM's technology treadmill. In fact, expect to see 256-Mbit flash memory by the end of the '90s.

FLASH MEMORY IS FAST

Don't be misled by technology-to-technology speed comparisons. Designing your system memory around flash will break the code/data bottleneck created by connecting a mechanical memory such as disks to a high-performance electronic system. For instance, data seek time for a 1.8″ magnetic hard disk is 20 ms, plus an 8 ms average rotational delay, while flash is less than 0.1 ms. At the chip level, current read speeds for flash are about 90 ns. Thus, downloading from flash to system RAM or directly executing from flash will dramatically enhance system speed.

FLASH MEMORY IS INEXPENSIVE

At the 8-Mbit density, Intel flash pricing matches DRAM and Intel expects to continue decreasing price as both densities and volumes increase.

FLASH IS NONVOLATILE

Unlike SRAM or pseudo-SRAM (SRAM with built-in battery), flash needs no battery backup. Further, Intel's flash devices retain data typically for over 100 years, well beyond the useful lifetime of even the most advanced computer.

FLASH IS POWER CONSCIOUS

FlashFile Memory in a hard-disk drive configuration consumes less than one two-hundredth the average power of a comparable magnetic disk drive based on the typical user model. At the chip level, the 28F008SA has a DEEP POWERDOWN mode that reduces power consumption to less than 0.2 μA.

FLASH IS RUGGED AND RELIABLE

On average, today's hard-disk drives can withstand up to 10 Gs of operating shock; Intel FlashFile memory can withstand as much as 1000 Gs. FlashFile compo-

nents can operate at up to 70°C while magnetic drives are limited to 55°C. Intel FlashFile memory can be cycled 100,000 times per block or segment. By employing wear-leveling techniques, a 20-Mbyte flash array can provide over 30 million hours before failure.

WHY NOW?

Flash memory is not a new technology. Intel has been the flash technology and market leader since 1988. Then why hasn't flash taken the mobile PC market by storm yet? Why now?

One reason that 1992 is the pivotal year for flash-based systems is the sharply increased demand for highly mobile computers. The other reason is that a number of key capabilities, in development for some time, reached maturity together.

1. Intel Introduces FlashFile™ Memory

MS-DOS*, the ubiquitous operating system for PCs, was developed specifically to optimize the sectoring scheme inherent to disk technology. Intel's first generation "bulk-erase" flash required that all of the chip be erased before data could be re-written: a natural fit for updatable firmware and data acquisition, but not for data file storage or disk emulation. Intel FlashFile memory, based on a block-erase architecture, divides the flash memory space into segments that are somewhat analogous to the zones recognized by MS-DOS. For instance, the Intel 28F008SA contains sixteen identical, individually-erasable, 64-Kbyte blocks. This organization has been carefully optimized to maximize cycling capability while preserving the smallest granularity possible. The ability to segment block memory into individual segments allows disk-like data-file storage.

2. Standardization of Delivery System and Interface

Thanks to work by the Personal Computer Memory Card International Association (PCMCIA), and the Japanese Electronics Industry Development Association (JEIDA), there is now an internationally recognized standard for memory cards. PCMCIA cards are the size of a business card but about four times as thick. Intel is widely promulgating its Exchangeable Card Architecture (ExCA™), a hardware and software implementation of the PCMCIA system interface. When used with the proper BIOS, ExCA/PCMCIA-compatible cards will be completely interchangeable between systems and vendors, and can be equated to solid-state floppy disks, albeit with many advantages.

Flash-based solid-state disks, intended to replace magnetic hard disks in certain applications, with IDE interfaces will be "plug compatible" with existing systems that are already designed with IDE magnetic drives in mind.

3. Flash File System

Intel has worked very closely with Microsoft* to implement a DOS flash memory extension called Flash File System (FFS) that transparently handles swapping of data between flash blocks, much as DOS now handles swaps between disk sectors. With Flash File System, the user inputs a DOS comand and doesn't need to think about whether a magnetic disk or a flash memory is being used. Flash File System employs wear leveling algorithms that prevent any block from being cycled excessively, thus ensuring millions of hours of use across multiple chips.

4. Off-the-Shelf Hardware Interface

The introduction of the Intel 82365SL PC Card Interface Controller provides a ready-made interface between the PC's ISA bus and up to two PCMCIA sockets. It is a key component for memory and I/O card implementations since the designer is relieved from building the interface from scratch.

5. Cost Reductions

Magnetic drives do not scale well; that is, it becomes increasingly difficult to improve or even retain density as platter size shrinks. Thus, every reduction in drive size requires complete retooling and costly learning. Also, the complex controller circuitry provides a price floor under which magnetic drives cannot drop. Since flash is scalable, at some point in the near future, small magnetic drives are likely to become more expensive per megabyte than flash cards and are certain to have less capacity. But even today, the value of a particular memory technology is a result of more than just dollars per megabyte.

Notes market analysis expert Dataquest:

> "The question is, "Can you put a floppy disk drive in a palmtop PC to take advantage of the cost disparity (between disk and flash)?" The answer is, 'No.' There is not enough power (or space). The issue then, is not cost. Here, the removable storage medium dictates the product's capabilities and its success or failure in the marketplace. Without a memory card, a palmtop is nothing more than an electronic organizer. It is the memory card that transforms a palmtop into a full-fledged personal computer." . . . Nick Samaras, SAMS Newsletter.

All of the aforementioned features, Intel FlashFile memory's block-erase architecture, PCMCIA standards, ExCA, Flash File System, 82365SL ISA-PCMCIA interface controller and reduced costs, are deliverables ... now. And not a moment too soon based on the tremendous market opportunity created by the increasing demand for truly mobile computers. Dataquest predicts that the worldwide sale of portable PCs will increase from six million units in 1992 to nearly 30 million units in 1995. While laptop PCs are only expected to increase by about two million units, notebooks, pen-based, and handheld PCs will increase from three million units in 1992 to nearly 25 million in 1995, an eight-fold increase. This extraordinary growth will be greatly assisted by FlashFile memory.

ENABLING THE TRULY MOBILE COMPUTER

In the world of the desktop PC, DRAM is used for executable code storage and data manipulation. Since DRAM is volatile, if power is lost, both programs and data are lost, hence the need for a nonvolatile magnetic hard disk. With the addition of the hard disk, programs and data are stored on the hard disk and swapped in and out of DRAM as needed. Some part of the DRAM is reserved for use as a register to store temporary results during compute-intensive operations. Today's PCs are typically configured with 4 megabytes of DRAM and at least a 40-Mbyte disk.

FlashFile memory fully supports this system configuration when used simply as a magnetic drive replacement. Instructions and data are still swapped to DRAM but at a much faster rate. Plus, execution speed can be enhanced if the DRAM is replaced with SRAM.

In the solid-state computer, the "DRAM + magnetic hard drive" are replaced by a "flash memory + SRAM". The key to this architecture is the ability to eXecute-InPlace (XIP). Program instructions stored in the flash memory are read directly by the processor. Results are written directly to the flash memory. Compute-intensive operations that require the fastest memory and byte-alterability use high-speed SRAM or pseudo SRAM. Most of what we now think of as the "DRAM" is replaced by low-cost flash and only a relatively small part of the DRAM is replaced by SRAM.

The flash memory space is made even more storage-efficient through the use of compression techniques which offer at least 2:1 compression. For example, one 20-Mbyte flash card that uses 2:1 compression offers the same storage as a 40-Mbyte hard disk!

The advantages of a flash-based computer include:

- Blazingly fast speed
- Instant-on and instant-resume
- Ultra-light PC (2–4 lbs.)
- Very secure data retention
- Flexible firmware

As you can see, by changing the system memory architecture to a flash-based one, designers will be able to build a new generation of PCs that meets the needs of the computer user of both today and tomorrow.

Progress has been made toward implementing this approach with the introduction of Hewlett-Packard's successful HP95LX DOS-compatible palmtop. MS-DOS and Lotus 1-2-3* are stored in ROM. Internally, pseudo-static RAM is used, and a PCMCIA memory socket is provided. Lotus 1-2-3 was re-written to allow ROM-based storage so it could execute in place. Other ROM-executable versions of popular operating systems are expected to be available shortly.

DESIGNING YOUR SYSTEM WITH FLASHFILE MEMORY

Details of the three Intel flash applications and implementations—flash cards, silicon disks, and Resident Flash Array (RFA)—are presented below.

APPLICATION NUMBER 1: MEMORY CARDS

Memory cards are the most rugged and reliable of the removable memory media. A card can be slipped into a shirt pocket and moved from location to location. With high-density flash cards, you can download files from the desktop and use the card in your notebook or palmtop.

Memory cards have been around for some time. The first cards to be introduced were ROM-only cards used in video games and pocket organizers. These were produced in various formats prior to the formation of PCMCIA and JEIDA. Later cards included battery-backed SRAM and EEPROM. Neither became very popular due to their high cost of $500–$600 per megabyte and limited capacity. Flash cards overcome the cost barrier and they are certain to be multiply sourced, assuring availability and competitive pricing. A 20-Mbyte flash card has three times the real storage density of a 20-Mbyte 1.8″ magnetic drive (0.95 Mbyte/cm^3 vs 0.34 Mbyte/cm^3) and it has 10 times the weight density (2 Mbyte/gm vs 0.21 Mbyte/gm). The PCMCIA has complete industry support, and enhanced versions, such as PCMCIA Version 2.0, are designed to be backward-compatible with earlier versions.

As part of its flash product family, Intel's new Series 2 memory cards are the first to utilize chips processed on its 0.8-micron flash technology. Storing up to 20 megabytes, these cards are designated Series 2 to differentiate them from the earlier bulk-erase flash cards. The cards consist of 4 to 20, 28F008SA TSOP FlashFile memory devices. Each 28F008SA contains 16 distinct, individually-erasable, 64-Kbyte blocks. Therefore, each card contains from 64 to 320 blocks.

With the release of PCMCIA Version 2.0 in September of 1991, the PCMCIA-compatible field grew somewhat larger. The PCMCIA interface grew from memory-only to one that supports many types of I/O devices. Intel's system-level implementation of PCMCIA 2.0, called ExCA, ensures that if there are two ExCA sockets available, one can be used for a flash memory card and one for a modem; and the cards may be interchanged.

How difficult is it to design-in an ExCA socket? Not very. Intel's open ExCA specification details the system implementation. Other than the physical incorporation of the socket and card, the only required hardware is an ISA-to-PCMCIA interface such as Intel's 82365SL chip, and an ExCA compliant BIOS from vendors like SystemSoft, Award and Phoenix. You'll also need a flash file management system like Microsoft's Flash File System. Intel's block-erase architecture, along with the DOS Filing System and ExCA BIOS, makes it easy to incorporate ExCA features. In addition, ExCA-compliant systems will allow system-to-system interoperability much like floppy disks.

APPLICATION NUMBER 2: FLASH-BASED SOLID STATE DISK

The implementation of block-erase flash as a "solid-state disk" (SSD) is something of a misnomer. It is not a disk at all, rather a flash module that has the same form, fit and function as a 2.5″ or smaller magnetic drive.

A flash-based SSD implementation is the most direct route to adapting flash to an existing design. A built-in IDE interface would make it plug-compatible. But what a difference a silicon disk will make! A 1.8″ drive typically uses one watt-hour/hour while a silicon disk uses as little as 0.035 watt-hour/hour. This kind of power savings makes it possible to reduce battery size and weight considerably. Or, consider reliability. We've already discussed differences in susceptibility to shock and temperature extremes. In addition, an SSD theoretically has a mean-time-between-failure (MBTF) of 250,000 hours, compared to 100,000 hours for the magnetic drive.

With all these advantages, when should you use memory cards and when is use of a flash-based SSD preferable?

First and foremost, the SSD is considered to be installable while memory cards are removable and transportable. In other words, the SSD is meant to be installed and then left alone, while memory cards are designed for constant removal and reinsertion. In operation, the only change a user would notice in a notebook computer equipped with a flash-based SSD is that access speed is unprecedented.

The flash-based solid-state drive is one very good way to get to market early with flash technology. In February 1992, Conner Peripherals, Inc., and Intel announced the signing of a joint product and technology development contract focused on designing and bringing to market proprietary FlashFile memory-based SSD storage products.

Incidentally, manufacturers of magnetic drives are starting to take notice. In a manner much like the tail wagging the dog, 1.8" magnetic hard disks with PCMCIA interfaces are currently being developed.

APPLICATION NUMBER 3: RESIDENT FLASH ARRAY

The one approach that offers totally new capabilities is the Resident Flash Array (RFA). This is an arrangement of from 8 to 20, 8-Mbit FlashFile memories. In the long term, it replaces some of the motherboard's DRAM. This is the approach that is applicable to all levels of PC, from desktop to palmtop. For near-term applications, however, RFA is an ideal way of making code or ROM-executable operating systems such as DOS or Windows* updatable to protect the end-user's software investment. Also, when used as a resident application program and data-file storage medium on the local memory bus, RFA provides a high-performance, low-power solution.

The Resident Flash Array provides the highest possible performance of any option, especially since the processor can be closely coupled to it; and hence, would not be encumbered by IDE or PCMCIA interfaces, or even the ISA bus itself. The flash memory and the processor will sit side-by-side.

The proliferation of flash memory card-based systems will accelerate the process of converting disk-oriented applications to a flash-executable orientation. Those manufacturers who elect to be early adopters of Intel FlashFile memory will be able to develop a new generation of PC—the truly "personal" computer you can hold in your hand.

NOTE:

ETOX, ExCA and FlashFile are trademarks of Intel Corporation.

*Microsoft and MS-DOS are registered trademarks; Windows is a trademark of Microsoft Corporation.

*Lotus and 1-2-3 are registered trademarks of Lotus Development Corporation.

intel®

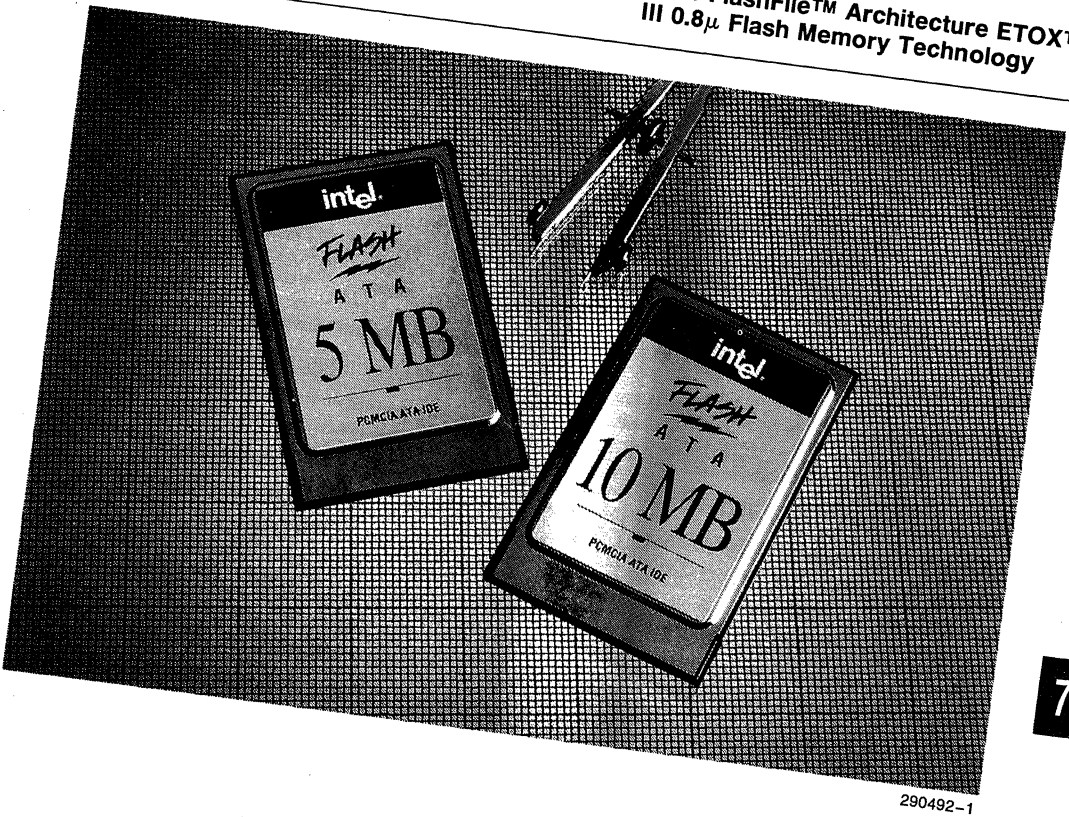
7

PCMCIA-ATA Flash Memory Drives

ADVANCE INFORMATION

iFD005P2SA/iFD010P2SA FLASH DRIVE

- PCMCIA-ATA Type-2 PC Card
- 10 mW Idle, Standby, and Sleep Power
- 5 MB/s Interface Transfer Rate
- 1000G Shock
- Four PCMCIA-ATA Modes
- IDE-ATA Mode
- Absolutely Silent
- Compatible with Standard Data Compression Utilities
- 5-MB and 10-MB Versions
- Embedded Flash Memory Cycle Management
- Weighs Only 1 oz. (29 gms)
- Nonvolatile; No Batteries Required
- Single +5V Supply
- 32-KB Buffer
- Uses FlashFile™ Architecture ETOX™ III 0.8μ Flash Memory Technology



290492-1

7

ETOX and FlashFile are trademarks of Intel Corporation.

November 1994
Order Number: 290492-002

iFD005P2SA/iFD010P2SA Flash Drive

CONTENTS

PAGE

INTRODUCTION 7-4

SPECIFICATION SUMMARY 7-5

PHYSICAL CHARACTERISTICS 7-6

RELIABILITY 7-6
Non-Recoverable Error Rate 7-6
Mean Time Between Failures (MTBF) 7-6
Error Correction 7-6
Preventive Maintenance 7-6

LOW-POWER APPLICATIONS 7-6

PCMCIA-STANDARD COMPATIBILITY 7-6

ATA-STANDARD AND COMPATIBILITY 7-6

CONFIGURATIONS 7-7

PHYSICAL DIMENSIONS 7-7

FLASH DRIVE PINOUT 7-8

FLASH DRIVE PIN DESCRIPTION 7-9

DATA ACCESS MODE TRUTH TABLES 7-11
Common Memory Plane 7-11
Attribute Memory Plane 7-11
IDE-ATA Mode Primary/Secondary 7-11
PCMCIA-ATA Mode — Independent Memory 7-12
PCMCIA-ATA Mode — Independent I/O 7-14
PCMCIA-ATA Mode — Primary/ Secondary 7-16

CONTENTS

PAGE

ADDRESSING MODES 7-18
IDE-ATA Mode 7-18
PCMCIA-ATA Independent Memory Mode 7-18
PCMCIA-ATA Independent I/O Mode 7-18
PCMCIA-ATA Primary or Secondary Mode 7-18

PCMCIA FUNCTIONS 7-19
Attribute Memory Access 7-19
Card Information Structure Tuples 7-20
PCMCIA Card Configuration Registers ... 7-25
Configuration Option Register 7-25
Card Configuration and Status Register 7-25
Pin Replacement Register 7-26
Socket and Copy Register 7-26

ATA FUNCTIONS
AT-Attachment Compatibility 7-26
Register Description 7-26
Data Register (Read/Write) 7-26
Error Register (Read) 7-26
Set Feature Register (Write) 7-27
Sector Count Register (Read/Write) .. 7-27
Sector Number Register (Read/Write) 7-27
Cylinder Low Register (Read/Write) .. 7-27
Cylinder High Register (Read/Write) 7-27
Drive/Head Register (Read/Write) 7-27
Status Register (Read) 7-27
Command Register (Write) 7-28

iFD005P2SA/iFD010P2SA Flash Drive

CONTENTS

PAGE

Alternate Status Register (Read) 7-28
Drive Control Register (Write) 7-28
Drive Address Register (Read) 7-28
ATA Command Descriptions 7-28
Recalibrate, 1X 7-30
Read Sector(s), 20 or 21 7-30
Read Long, 22 or 23 7-30
Write Sector(s), 30 or 31 7-30
Write Long, 32 or 33 7-30
Read Verify Sector(s), 40 or 41 7-30
Format Track, 50 7-30
Seek, 7X 7-30
Execute Drive Diagnostic, 90 7-30
Initialize Drive Parameters, 91 7-31
Standby Immediate, 94 or E0 7-31
Idle Immediate, 95 or E1 7-31
Standby, 96 or E2 7-31
Idle, 97 or E3 7-31
Check Power Mode, 98 or E5 7-31
Set Sleep Mode, 99 or E6 7-31

CONTENTS

PAGE

Intel Get/Set Features, 9A 7-31
Read Multiple, C4 7-31
Write Multiple, C5 7-32
Set Multiple Mode, C6 7-32
Read DMA, C8 or C9 7-32
Write DMA, CA or CB 7-32
Read Buffer, E4 7-32
Write Buffer, E8 7-32
Identify Drive, EC 7-32
Set Features, EF 7-33
Error Posting 7-33

ABSOLUTE MAXIMUM RATINGS 7-36

DC CHARACTERISTICS 7-36

TIMING INFORMATION 7-37

INDEX 7-41

ORDERING INFORMATION 7-45

REVISION HISTORY 7-45

INTRODUCTION

Intel's Flash Drive brings FlashFile memory's mass storage advantages to PCMCIA-ATA and IDE-ATA-equipped mobile computers. These nonvolatile storage products provide performance, ruggedness, reliability, silent operation, and extended battery life that these systems need.

In default mode, the flash drive operates in **PCMCIA-ATA** compliant sockets. It conforms to PCMCIA PC Card standard 2.1, (with ATA version 1.02 hardware and software).

Its **IDE-ATA auto-detect mode** allows it to operate in standard IDE hardware- and BIOS-equipped systems (with 68-pin connector). It uses the industry-standard ATA command set so no software drivers are required.

PCMCIA and ATA power management commands are supported. In addition, **automatic power management** reduces power without system intervention. Its instant-on capability allows it to enter a power-down mode when not active; this reduces power to 10 mW or less.

Its proprietary energy-saving modes can double system battery life. Unlike disk drives that consume substantial disk-spinning and head-positioning power, the flash drive's solid state circuitry uses power only when managing and transferring data.

Spinup, seeks, and rotational latency, which substantially degrade hard disk drive performance, are virtually non-existent in the flash drive. This, coupled with 5 MB/s interface and 8 MB/s read-media transfer rates, makes it a very **high performance ATA drive.**

The PCMCIA Type-2 profile maintains interface compatibility with Type-3 1.8" PCMCIA- and IDE-ATA magnetic hard disk drives; at only **half the thickness and half the weight.** And, its **solid-state construction, embedded flash memory management, 1000G shock specification, built-in ECC, and flash memory's inherent reliability** provide maximum data integrity.

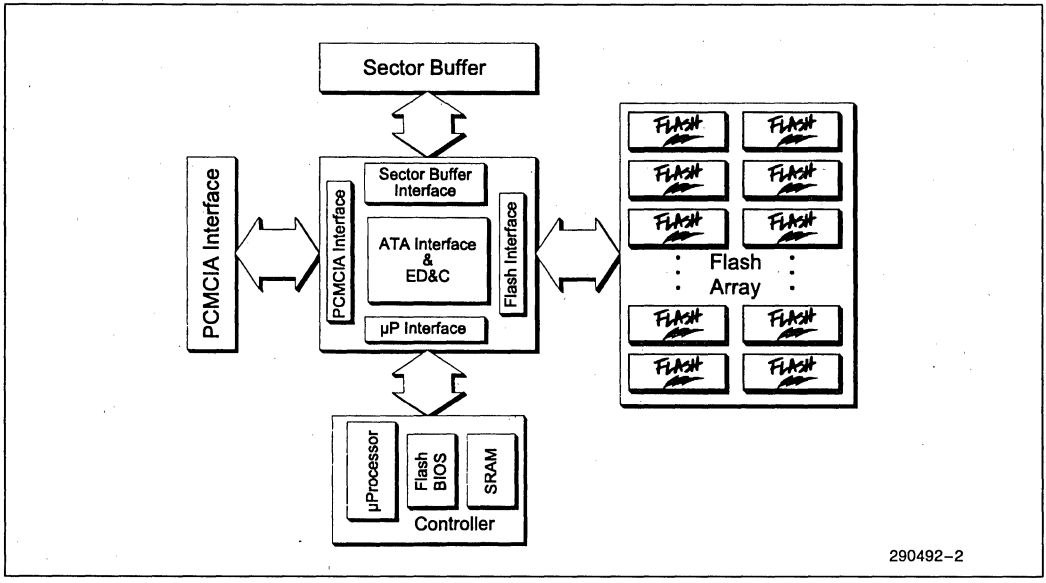


Figure 1. The Flash Drive's Block Diagram

SPECIFICATION SUMMARY

Capacity (CHS Auto-Translate)

Formatted	Bytes (cylinders, heads, sectors)
iFD005, 5M	5,111,808 (156, 2, 32)
iFD010, 10M	10,485,760 (320, 2, 32)

Environmental

Temperature	
Operating	0°C to 60°C
Non-operating	−30°C to +70°C
Humidity, Operating (non-cond, 26°C wet bulb)	5% RH to 95% RH
Altitude, Operating	−60m to +12 Km (−200 ft to +40K ft)

Interfaces

68-pin PCMCIA-ATA
PCMCIA PC Card Standard, Release 2.1
PCMCIA PC Card ATA Specification 1.02

68-pin IDE-ATA 1.8″ disk-drive-type ATA
ATA Interface for Disk Drives Standard 4A

Performance

Seek Time (maximum or track-to-track)	<1 ms
Rotational Latency	0 ms
Spinup Time	
Via Command (from deep power-down)	10 ms
Power-on to ready	2.5s (max)
Media Transfer Rate	
Read	8.0 MB/s
Write	0.27 MB/s
Interface Transfer Rate (burst)	5.0 MB/s
Buffer Size	32 KB

Power (see DC specifications for details)

Read	0.65W
Write	0.7W
Idle	0.010W
Standby	0.010W
Sleep mode	0.010W
Peak current (programming voltage start-up)	0.7A
Supply Voltage	5.0V ±5%

Reliability

MTBF (min)	250K power-on-hours
(typical)	5M power-on-hours
(T_A = 25°C, 25 KB write every 5 minutes)	
Read Error Rate (with retries and ECC)	1 in 10^{14} bits read
ECC (optimized for flash memory)	32 bits/sector

Ruggedness (any axis or direction)

Shock, op/non-operating	1000G
Vibration, op/non-operating	>15G, 10 Hz–2000 Hz

Size 85.6mm x 54.0mm x 5.0mm (3.37″ x 2.126″ x 0.196″)

Weight 29 gms (1.05 oz.)

Noise Absolutely silent operation (Sound Pressure Level) 0 SPL

7

PHYSICAL CHARACTERISTICS

Intel's Flash Drive has one surface-mount circuit board. Dimensions and physical characteristics are PCMCIA and industry-standard 68-pin IDE-ATA disk drive compatible.

RELIABILITY

Non-Recoverable Error Rate

Minimum error rate is 1 error per 10^{14} bits read with retries and error correction enabled (typically 1 error in 10^{17}). Correctable defects are not included.

Mean Time Between Failures (MTBF)

Mean Time Between Failure (MTBF) estimates the time between physical repair or faulty-part replacement to restore a unit to full functionality. It is the reciprocal of failure rate during useful life, when failures are random and the failure-occurrence rate is constant. Flash drive MTBF calculations incorporate flash cycling to give a clearer comparison to magnetic hard disk drives.

MTBF conditions:

Environment: (T_A = 25°C)

25 KB write every 5 minutes over 250,000 hours on 10-MB drive.

Error Correction

The 32-bit error correction code can correct one error burst (8 bits maximum) per 512-byte sector.

Preventive Maintenance

No preventive maintenance is required.

LOW-POWER APPLICATIONS

The flash drive's low-power modes make it ideal for battery-based applications.

- **Self-initiated deep power-down mode** places non-critical circuitry, including the flash array and programming power unit, into a power-down mode (which is transparent to the host) when the drive detects no activity. The drive powers up and responds to a host command within 10 ms.
- *Idle* or *standby* commands initiate **Idle or Standby modes.** If requested by the host, the drive places itself in idle mode for about 1.5 seconds. If no activity is detected, it transitions to standby mode, which is identical to its sleep mode (or self-initiated deep power-down mode). The drive wakes upon receiving any command.
- **Host-initiated Sleep mode** places all flash drive circuitry in a power-down state when the host executes a set sleep mode command. A hardware or software reset brings the drive out of sleep mode.

PCMCIA-STANDARD COMPATIBILITY

Intel's Flash Drive is compatible with PCMCIA PC Card Standard, Release 2.1. To obtain this specification, write to:

PCMCIA
1030G East Duane Avenue
Sunnyvale, Ca 94086
Tel: (408) 720-0107
Fax: (408) 720-9416

ATA-STANDARD AND COMPATIBILITY

The flash drive is compatible with the "AT-Attachment Interface for Disk Drives" draft proposal Revision 4A. *See the internal working document X3T9.2, a Task Group of Accredited Standards Committee X3 of the American National Standard for Information Systems and the AT Attachment Specification for detailed information.* To obtain this specification, write to:

AT-Attachment Document Distribution
Global Engineering
15 Inverness Way East
Englewood, Co. 80112-5704
Tel: (800) 854-7179 or (303)792-2181
Fax: (303) 792-2192

CONFIGURATIONS

The flash drive has five addressing configurations that allow system designers to tailor hardware and software for system requirements. These configurations, described later, are:

1.8" AT-Attachment (IDE) Mode

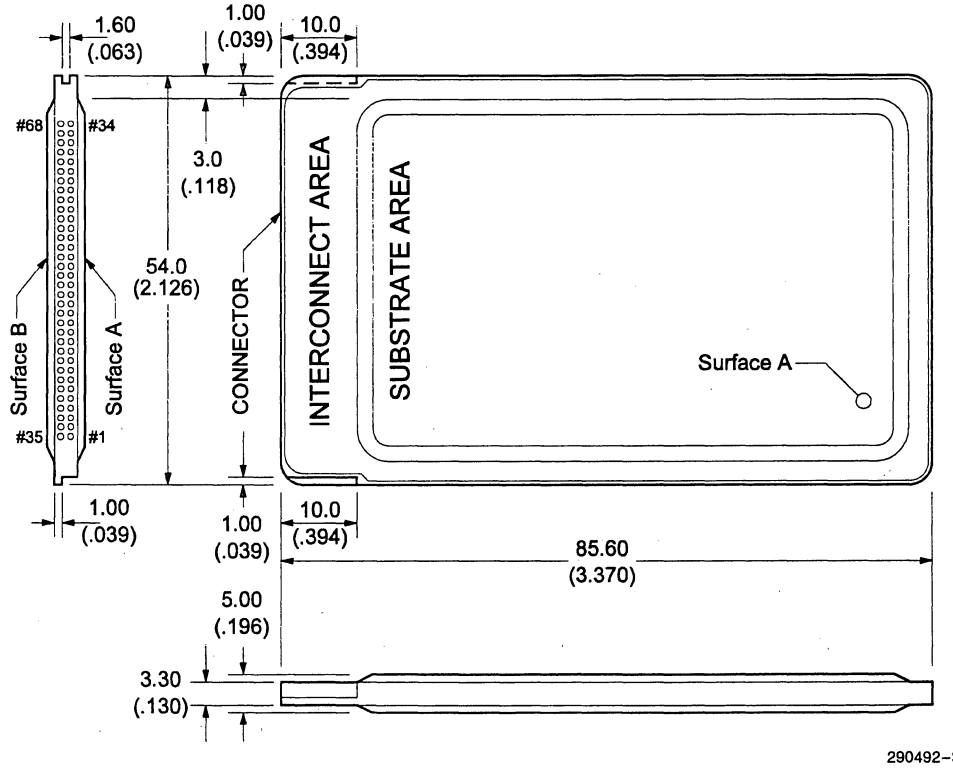
- 68-pin IDE-ATA disk drive emulation.

PCMCIA-ATA Modes

- Memory Mode (2-KByte contiguous memory space)
- Independent I/O (16 contiguous I/O addresses)
- Primary Drive Address (1F0–1F7, 3F6 and 3F7)
- Secondary Drive Address (170–177, 376 and 377).

PHYSICAL DIMENSIONS

Figure 2 shows the drive's 68-pin PCMCIA Type-2 memory-card case dimensions.



NOTE:
(Inches in parenthesis)

Figure 2. Flash Drive Dimensions

7

Table 1. Flash Drive Pinout

Pin	PCMCIA Memory, I/O Signal	PCMCIA-ATA Mode	IDE-ATA Mode	Notes
1	GND	GND	GND	
2	D3	D3	DD3	
3	D4	D4	DD4	
4	D5	D5	DD5	
5	D6	D6	DD6	
6	D7	D7	DD7	
7	CE1#	CE1#	CS1FX#	
8	A10	A10	N.U.	2
9	OE#	OE#	SELATA#	5
10	A11	N.C.	N.C.	1
11	A9	A9	N.U.	2
12	A8	A8	N.U.	2
13	A13	N.C.	N.C.	1
14	A14	N.C.	N.C.	1
15	WE#/PGM#	WE#	N.U.	2
16	RDY/BSY#/ IREQ#	RDY/BSY#/ IREQ#	INTRQ	4
17	V_{CC}	V_{CC}	V_{CC}	
18	V_{PP1}	N.C.	N.C.	1
19	A16	N.C.	N.C.	1
20	A15	N.C.	N.C.	1
21	A12	N.C.	N.C.	1
22	A7	A7	N.U.	2
23	A6	A6	N.U.	2
24	A5	A5	N.U.	2
25	A4	A4	N.U.	2
26	A3	A3	N.U.	2
27	A2	A2	DA2	
28	A1	A1	DA1	
29	A0	A0	DA0	
30	D0	D0	DD0	
31	D1	D1	DD1	
32	D2	D2	DD2	
33	WP/IOIS16#	IOIS16#	IOCS16#	
34	GND	GND	GND	

Pin	PCMCIA Memory, I/O Signal	PCMCIA-ATA Mode	IDE-ATA Mode	Notes
35	GND	GND	GND	
36	CD1#	CD1#	N.U.	2
37	D11	D11	DD11	
38	D12	D12	DD12	
39	D13	D13	DD13	
40	D14	D14	DD14	
41	D15	D15	DD15	
42	CE2#	CE2#	CS3FX#	
43	RFSH#	N.C.	N.C.	1
44	IORD#	IORD#	DIOR#	
45	IOWR#	IOWR#	DIOW#	
46	A17	N.C.	N.C.	1
47	A18	N.C.	N.C.	1
48	A19	N.C.	N.C.	1
49	A20	N.C.	N.C.	1
50	A21	N.C.	N.C.	1
51	V_{CC}	V_{CC}	V_{CC}	
52	V_{PP2}	N.C.	N.C.	1
53	A22	N.C.	N.C.	1
54	A23	N.C.	N.C.	1
55	A24	N.C.	N.C.	1
56	A25	N.C.	N.C.	1
57	Reserved	N.C.	N.C.	1
58	RESET	RESET	RESET#	4
59	WAIT#	WAIT#	IORDY	
60	INPACK#	INPACK#	DMARQ	
61	REG#	REG#	DMACK#	
62	BVD2	Pullup	DASP#	
63	BVD1/ STSCHG#	STSCHG#	PDIAG#	
64	D8	D8	DD8	
65	D9	D9	DD9	
66	D10	D10	DD10	
67	CD2#	CD2#	N.U.	2
68	GND	GND	GND	

NOTES:
1. N.C. = Not connected internally.
2. N.U. = Connected internally but Not Used in mode.
3. D.U. = Don't Use in specified mode; internally driven.
4. PCMCIA-ATA RESET and IREQ# polarities are RESET# and INTRQ in IDE-ATA mode.
5. OE# (SELATA#) is used during reset to determine PCMCIA-ATA or IDE-ATA (if grounded) configuration.

Table 2. Flash Drive Pin Description

Symbol	Pin	Direction*	Description
GND	1,34,35, 68		Ground between the host and drive.
D0–15	2–6, 30–32, 37–41, 64–66	I/O	**PCMCIA-ATA MODE:** 16-bit bi-directional data bus between the host and drive.
DD0–15			**IDE-ATA MODE:** Register and ECC accesses use DD0–7. Data transfers use DD0–15.
CE1#	7	I	**PCMCIA-ATA MODE:** Card Enable 1 enables even data bytes on D0–7 (see Table 3 for detailed description).
CS1FX#			**IDE-ATA MODE:** IDE-ATA Host Card Select 0 accesses command block registers.
A0–25	8, 10–14, 19-29, 46-50, 53-56	I	**PCMCIA-ATA MODE:** Addresses A0–10 access data and registers depending on the memory or I/O mode chosen by the host (see Table 3). Addresses A11–24 are not used.
DA0–2			**IDE-ATA MODE:** Only addresses DA0–2 are used (see Table 3).
OE#	9	I	**PCMCIA-ATA MODE:** Output Enable used to read attribute- and memory-mode data onto D0-15.
SELATA#			**IDE-ATA MODE:** Detects IDE-ATA mode when externally grounded during reset (see Table 3).
WE#	15	I	**PCMCIA-ATA MODE:** Active-low Write Enable used to write attribute- and memory-mode data that is on D0-15.
			IDE-ATA MODE: Not used.
RDY/BSY#/ IREQ#	16	O	**PCMCIA-ATA MODE:** Indicates internally timed activities status. Drive can accept host accesses when high. In memory mode, RDY/BSY# does not reflect the ATA status register's BSY bit.
INTRQ			**IDE-ATA MODE:** Host enables Interrupt Request only when drive is selected and the host clears the Drive Control register's IEN#. INTRQ (a three-state pin) is high-Z when IEN# is set or the drive is not selected. INTRQ is set when the drive's CPU sets IRQ. INTRQ resets during Status register read or Command register write.
V_{CC}	17, 51	I	+5V DC supply to the drive.
V_{PP1},V_{PP2}	18, 52	N.C.	+12V DC programming supply is not required.
IOIS16#	33	O	**PCMCIA-ATA MODE:** Held low after the reset initialization sequence (Write Protect is not supported).
IOCS16#			**IDE-ATA MODE:** Host I/O 16 tells the host that the data register was accessed and the drive can send/receive 16-bit data. IOCS16# is a three-state pin.

***NOTE:**
All pin directions are referenced to the drive.

Table 2. Flash Drive Pin Description (Continued)

Symbol	Pin	Direction*	Description
CD1#, CD2#	36, 67	O	Card Detect pins, internally pulled to ground, allow the host to determine that the drive is fully inserted in the socket.
CE2#	42	I	**PCMCIA-ATA MODE:** Card Enable 2 enables odd data bytes on D8-15 (see Table 3).
CS3FX#			**IDE-ATA MODE:** Host Card Select 1 selects drive control block registers.
RFSH#	43	N.C.	Not used.
IORD#	44	I	**PCMCIA-ATA MODE:** Active-low I/O Read gates data onto D0–15 during I/O-mode accesses.
DIOR#			**IDE-ATA MODE:** Host I/O Read enables drive register data onto DD0-15. The host latches drive data on DIOR#'s rising edge.
IOWR#	45	I	**PCMCIA-ATA MODE:** Active-low I/O Write gates data from D0–15 during I/O-mode accesses.
DIOW#			**IDE-ATA MODE:** Host I/O Write's rising edge strobes data into a drive register.
Reserved	57	N.C.	Reserved.
RESET	58	I	**PCMCIA-ATA MODE:** At power-on or hardware reset, this active-high signal resets all drive registers.
RESET#			**IDE-ATA MODE:** At power-on or hardware reset, this active-low signal resets all drive registers.
WAIT#	59	O	**PCMCIA-ATA MODE:** The drive's WAIT# signals the host that the in-progress memory or I/O cycle is not complete.
IORDY			**IDE-ATA MODE:** IORDY is not implemented in IDE-ATA mode.
INPACK#	60	O	**PCMCIA-ATA MODE:** When selected, the drive asserts Input Acknowledge while responding to an I/O read cycle.
DMARQ			**IDE-ATA MODE:** Host DMA Request is not supported.
REG#	61	I	**PCMCIA-ATA MODE:** Common memory is accessed when high. Attribute memory and ATA I/O registers are accessed when low.
DMACK#			**IDE-ATA MODE:** Host DMA ACKnowledge is not supported.
BVD2	62	O	**PCMCIA-ATA MODE:** 10K Pull-up to V_{CC}.
DASP# #			**IDE-ATA MODE:** Indicates Drive-Active or Second-Present when the disk is accessed. This open-drain output is pulled up with a 10 KΩ resistor.
BVD1/ STSCHG#	63	I/O	**PCMCIA-ATA MODE:** Notifies the host of RDY/BSY# and Write Protect state changes.
PDIAG#			**IDE-ATA MODE:** In slave mode (not supported), Passed Diagnostics (low) tells a master that diagnostics was passed. In master mode, it is an input.

***NOTE:**
All pin directions are referenced to the drive.

Table 3. Data Access Mode Truth Table

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–D8	D7–D0
COMMON MEMORY PLANE											
Standby	A0 = X(1)		X	H	H	X	X	X	X	High-Z	High—Z
Byte	A0 = L	Read	H	H	L	L	H	H	H	High-Z	Even-Byte
	A0 = H		H	H	L	L	H	H	H	High-Z	Odd-Byte
Odd-Byte	A0 = X		H	L	H	L	H	H	H	Odd-Byte	High-Z
Word	A0 = X		H	L	L	L	H	H	H	Odd-Byte	Even-Byte
Byte	A0 = L	Write	H	H	L	H	L	H	H	X	Even-Byte
	A0 = H		H	H	L	H	L	H	H	X	Odd-Byte
Odd-Byte	A0 = X		H	L	H	H	L	H	H	Odd-Byte	X
Word	A0 = X		H	L	L	H	L	H	H	Odd-Byte	Even-Byte
ATTRIBUTE MEMORY PLANE											
Standby	A0 = X		X	H	H	X	X	X	X	High-Z	High-Z
Byte	A0 = L	Read	L	H	L	L	H	H	H	High-Z	Even-Byte
	A0 = H		L	H	L	L	H	H	H	High-Z	Invalid
Odd-Byte	A0 = X		L	L	H	L	H	H	H	Invalid	High-Z
Word	A0 = X		L	L	L	L	H	H	H	Invalid	Even-Byte
Byte	A0 = L	Write	L	H	L	H	L	H	H	X	Even-Byte
	A0 = H		L	H	L	H	L	H	H	X	Invalid
Odd-Byte	A0 = X		L	L	H	H	L	H	H	Invalid	X
Word	A0 = X		L	L	L	H	L	H	H	Invalid	Even-Byte
IDE-ATA MODE—Primary = 1F0–1F7, 3F6–3F7; Secondary = 170–177, 376–377(2)											
No Operation	X		X	H	H	X	X	X	X	High-Z	High-Z
Invalid	X		H	L	L	L	X	X	X	Invalid	Invalid
Data	0	Read	H	H	L	L	H	L	H	Data High	Data Low
		Write	H	H	L	L	H	H	L	Data High	Data Low
Error	1	Read	H	H	L	L	H	L	H	Invalid	Error
Set Feature		Write	H	H	L	L	H	H	L	Invalid	Feature
Sector Count	2	Read	H	H	L	L	H	L	H	Invalid	Sect. Cnt.
		Write	H	H	L	L	H	H	L	Invalid	Sect. Cnt.
Sector Number	3	Read	H	H	L	L	H	L	H	Invalid	Sect. No.
		Write	H	H	L	L	H	H	L	Invalid	Sect. No.

7

Table 3. Data Access Mode Truth Table (Continued)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–D8	D7–D0
IDE-ATA MODE—Primary = 1F0–1F7, 3F6–3F7; Secondary = 170–177, 376–377[2] (Continued)											
Cylinder Low	4	Read	H	H	L	L	H	L	H	Invalid	Cyl. Low
		Write	H	H	L	L	H	H	L	Invalid	Cyl. Low
Cylinder High	5	Read	H	H	L	L	H	L	H	Invalid	Cyl. High
		Write	H	H	L	L	H	H	L	Invalid	Cyl. High
Drive/Head	6	Read	H	H	L	L	H	L	H	Invalid	Drv/Hd
		Write	H	H	L	L	H	H	L	Invalid	Drv/Hd
Status	7	Read	H	H	L	L	H	L	H	Invalid	Status
Command		Write	H	H	L	L	H	H	L	Invalid	Command
Invalid	0–5		H	L	L	L	X	X	X	Invalid	Invalid
Alternate Status	6	Read	H	L	H	L	H	L	H	Invalid	Alt. Status
Drive Control		Write	H	L	H	L	H	H	L	Invalid	Control
Drive Address	7	Read	H	L	H	L	H	L	H	Invalid	Drv. Add.
		Write	H	L	H	L	H	H	L	Invalid	Not Used
PCMCIA-ATA MODE—Independent Memory Address = 000–00F, 400–7FF[3]											
No Operation	X		X	H	H	X	X	X	X	High-Z	High-Z
Data[6]	000	Read	H	H	L	L	H	H	H	Invalid	Data
			H	L	L	L	H	H	H	Data High	Data Low
		Write	H	H	L	H	L	H	H	X	Data
			H	L	L	H	L	H	H	Data High	Data Low
Error	001	Read	H	H	L	L	H	H	H	Invalid	Error
			H	L	H	L	H	H	H	Error	Invalid
Set Feature		Write	H	H	L	H	L	H	H	X	Feature
			H	L	H	H	L	H	H	Feature	X
Sector Count	002	Read	H	H	L	L	H	H	H	Invalid	Sect. Cnt.
			H	L	L	L	H	H	H	Sect. No.	Sect. Cnt.
		Write	H	H	L	H	L	H	H	X	Sect. Cnt.
			H	L	L	H	L	H	H	Sect. No.	Sect. Cnt.
Sector Number	003	Read	H	H	L	L	H	H	H	Invalid	Sect. No.
			H	L	H	L	H	H	H	Sect. No.	Invalid
		Write	H	H	L	H	L	H	H	X	Sect. No.
			H	L	H	H	L	H	H	Sect. No.	X

Table 3. Data Access Mode Truth Table (Continued)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–D8	D7–D0
PCMCIA-ATA MODE—Independent Memory Address = 000–00F, 400–7FF[3] (Continued)											
Cylinder Low	004	Read	H	H	L	L	H	H	H	Invalid	Cyl. Low
			H	L	L	L	H	H	H	Cyl. High	Cyl. Low
		Write	H	H	L	H	L	H	H	X	Cyl. Low
			H	L	L	H	L	H	H	Cyl. High	Cyl. Low
Cylinder High	005	Read	H	H	L	L	H	H	H	Invalid	Cyl. High
			H	L	H	L	H	H	H	Cyl. High	Invalid
		Write	H	H	L	H	L	H	H	X	Cyl. High
			H	L	H	H	L	H	H	Cyl. High	X
Drive/Head	006	Read	H	H	L	L	H	H	H	Invalid	Drv/Hd
			H	L	L	L	H	H	H	Status	Drv/Hd
		Write	H	H	L	H	L	H	H	X	Drv/Hd
			H	L	L	H	L	H	H	Command	Drv/Hd
Status	007	Read	H	H	L	L	H	H	H	Invalid	Status
			H	L	H	L	H	H	H	Status	Invalid
Command		Write	H	H	L	H	L	H	H	X	Command
			H	L	H	H	L	H	H	Command	X
Data (Duplicate)[6]	008	Read	H	H	L	L	H	H	H	Invalid	Data
			H	L	L	L	H	H	H	Data High	Data Low
		Write	H	H	L	H	L	H	H	X	Data
			H	L	L	H	L	H	H	Data High	Data Low
	009	Read	H	H	L	L	H	H	H	Invalid	Data High
			H	L	H	L	H	H	H	Data High	Invalid
		Write	H	H	L	H	L	H	H	X	Data High
			H	L	H	H	L	H	H	Data High	X
Invalid	00A–00C		H	X	X	X	X	X	X	Invalid	Invalid
Error (Duplicate)	00D	Read	H	H	L	L	H	H	H	Invalid	Error
			H	L	H	L	H	H	H	Error	Invalid
		Write	H	H	L	H	L	H	H	X	Feature
			H	L	H	H	L	H	H	Feature	X
Alternate Status	00E	Read	H	H	L	L	H	H	H	Invalid	Alt. Status
			H	L	L	L	H	H	H	Drv. Add.	Alt. Status
Drive Control		Write	H	H	L	H	L	H	H	X	Control
			H	L	L	H	L	H	H	Not Used	Control

7

Table 3. Data Access Mode Truth Table (Continued)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–D8	D7–D0
PCMCIA-ATA MODE—Independent Memory Address = 000–00F, 400–7FF[3] (Continued)											
Drive Address	00F	Read	H	H	L	L	H	H	H	Invalid	Drv. Add.
			H	L	H	L	H	H	H	Drv. Add.	Invalid
		Write	H	H	L	H	L	H	H	X	Not Used
			H	L	H	H	L	H	H	Not Used	X
Data (Duplicate)[6]	400–7FF A0 = 0 (Even address)	Read	H	H	L	L	H	H	H	Invalid	Data Low
			H	L	L	L	H	H	H	Data High	Data Low
		Write	H	H	L	H	L	H	H	X	Data Low
			H	L	L	H	L	H	H	Data High	Data Low
	400–7FF A0 = 1 (Odd address)	Read	H	H	L	L	H	H	H	Invalid	Data High
			H	L	H	L	H	H	H	Data High	Invalid
		Write	H	H	L	H	L	H	H	X	Data High
			H	L	H	H	L	H	H	Data High	X
PCMCIA-ATA MODE—Independent I/O Address = xx0–xxF[4]											
No Operation	X		X	H	H	X	X	X	X	High-Z	High-Z
Data	0	Read	L	H	L	H	H	L	H	Invalid	Data
			L	L	L	H	H	L	H	Data High	Data Low
		Write	L	H	L	H	H	H	L	X	Data
			L	L	L	H	H	H	L	Data High	Data Low
Error	1	Read	L	H	L	H	H	L	H	Invalid	Error
			L	L	H	H	H	L	H	Error	Invalid
Set Feature		Write	L	H	L	H	H	H	L	X	Feature
			L	L	H	H	H	H	L	Feature	X
Sector Count	2	Read	L	H	L	H	H	L	H	Invalid	Sect. Cnt.
			L	L	L	H	H	L	H	Sect. No.	Sect. Cnt.
		Write	L	H	L	H	H	H	L	X	Sect. Cnt.
			L	L	L	H	H	H	L	Sect. No.	Sect. Cnt.
Sector Number	3	Read	L	H	L	H	H	L	H	Invalid	Sect. No.
			L	L	H	H	H	L	H	Sect. No.	Invalid
		Write	L	H	L	H	H	H	L	X	Sect. No.
			L	L	H	H	H	H	L	Sect. No.	X

Table 3. Data Access Mode Truth Table (Continued)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–D8	D7–D0
PCMCIA-ATA MODE—Independent I/O Address = xx0–xxF[4] (Continued)											
Cylinder Low	4	Read	L	H	L	H	H	L	H	Invalid	Cyl. Low
			L	L	L	H	H	L	H	Cyl. High	Cyl. Low
		Write	L	X	L	H	H	H	L	X	Cyl. Low
			L	X	L	H	H	H	L	Cyl. High	Cyl. Low
Cylinder High	5	Read	L	H	L	H	H	L	H	Invalid	Cyl. High
			L	L	H	H	H	L	H	Cyl. High	Invalid
		Write	L	H	L	H	H	H	L	X	Cyl. High
			L	L	H	H	H	H	L	Cyl. High	X
Drive/Head	6	Read	L	H	L	H	H	L	H	Invalid	Drv/Hd
			L	L	L	H	H	L	H	Status	Drv/Hd
		Write	L	H	L	H	H	H	L	X	Drv/Hd
			L	L	L	H	H	H	L	Command	Drv/Hd
Status	7	Read	L	H	L	H	H	L	H	Invalid	Status
			L	L	H	H	H	L	H	Status	Invalid
Command		Write	L	H	L	H	H	H	L	X	Command
			L	L	H	H	H	H	L	Command	X
Data (Duplicate)	8	Read	L	H	L	H	H	L	H	Invalid	Data
			L	L	L	H	H	L	H	Data High	Data Low
		Write	L	H	L	H	H	H	L	X	Data
			L	L	L	H	H	H	L	Data High	Data Low
	9	Read	L	H	L	H	H	L	H	Invalid	Data High
			L	L	H	H	H	L	H	Data High	Invalid
		Write	L	H	L	H	H	H	L	X	Data High
			L	L	H	H	H	H	L	Data High	X
Invalid	A–C		L	X	X	X	X	X	X	Invalid	Invalid
Error (Duplicate)	D	Read	L	H	L	H	H	L	H	Invalid	Error
			L	L	H	H	H	L	H	Error	Invalid
		Write	L	H	L	H	H	H	L	X	Feature
			L	L	H	H	H	H	L	Feature	X

7

Table 3. Data Access Mode Truth Table (Continued)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–D8	D7–D0
PCMCIA-ATA MODE—Independent I/O Address = xx0–xxF(4) (Continued)											
Alternate Status	E	Read	L	H	L	H	H	L	H	Invalid	Alt. Status
			L	L	L	H	H	L	H	Drv. Add.	Alt. Status
Drive Control		Write	L	H	L	H	H	H	L	X	Control
			L	L	L	H	H	H	L	Not Used	Control
Drive Address	F	Read	L	H	L	H	H	L	H	Invalid	Drv. Add.
			L	L	H	H	H	L	H	Drv. Add.	Invalid
		Write	L	H	L	H	H	H	L	X	Not Used
			L	L	H	H	H	H	L	Not Used	X
PCMCIA-ATA MODE—Primary Address = 1F0–1F7, 3F6–3F7; Secondary Address = 170–177, 376–377(5)											
No Operation	X		X	H	H	X	X	X	X	High-Z	High-Z
Data	1F0/170	Read	L	H	L	H	H	L	H	Invalid	Data Low
			L	L	L	H	H	L	H	Data High	Data Low
		Write	L	H	L	H	H	H	L	X	Data Low
			L	L	L	H	H	H	L	Data High	Data Low
Error	1F1/171	Read	L	H	L	H	H	L	H	Invalid	Error
			L	L	H	H	H	L	H	Error	Invalid
Set Feature		Write	L	H	L	H	H	H	L	X	Feature
			L	L	H	H	H	H	L	Feature	X
Sector Count	1F2/172	Read	L	H	L	H	H	L	H	Invalid	Sect. Cnt.
			L	L	L	H	H	L	H	Sect. No.	Sect. Cnt.
		Write	L	H	L	H	H	H	L	X	Sect. Cnt.
			L	L	L	H	H	H	L	Sect. No.	Sect. Cnt.
Sector Number	1F3/173	Read	L	H	L	H	H	L	H	Invalid	Sect. No.
			L	L	H	H	H	L	H	Sect. No.	Invalid
		Write	L	H	L	H	H	H	L	X	Sect. No.
			L	L	H	H	H	H	L	Sect. No.	X
Cylinder Low	1F4/174	Read	L	H	L	H	H	L	H	Invalid	Cyl. Low
			L	L	L	H	H	L	H	Cyl. High	Cyl. Low
		Write	L	H	L	H	H	H	L	X	Cyl. Low
			L	L	L	H	H	H	L	Cyl. High	Cyl. Low

Table 3. Data Access Mode Truth Table (Continued)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–D8	D7–D0
PCMCIA-ATA MODE—Primary Address = 1F0–1F7, 3F6–3F7; Secondary Address = 170–177, 376–377[5] (Continued)											
Cylinder High	1F5/175	Read	L	H	L	H	H	L	H	Invalid	Cyl. High
			L	L	H	H	H	L	H	Cyl. High	Invalid
		Write	L	H	L	H	H	H	L	X	Cyl. High
			L	L	H	H	H	H	L	Cyl. High	X
Drive/Head	1F6/176	Read	L	H	L	H	H	L	H	Invalid	Drv/Hd
			L	L	L	H	H	L	H	Status	Drv/Hd
		Write	L	H	L	H	H	H	L	X	Drv/Hd
			L	L	L	H	H	H	L	Command	Drv/Hd
Status	1F7/177	Read	L	H	L	H	H	L	H	Invalid	Status
			L	L	H	H	H	L	H	Status	Invalid
Command		Write	L	H	L	H	H	H	L	X	Command
			L	L	H	H	H	H	L	Command	X
Invalid	3F/370–5	R/W	L	X	X	X	X	X	X	Invalid	Invalid
Alternate Status	3F6/376	Read	L	H	L	H	H	L	H	Invalid	Alt. Status
			L	L	L	H	H	L	H	Drv. Add.	Alt. Status
Drive Control		Write	L	H	L	H	H	H	L	X	Control
			L	L	L	H	H	H	L	Not Used	Control
Drive Address	3F7/377	Read	L	H	L	H	H	L	H	Invalid	Drv. Add.
			L	L	H	H	H	L	H	Drv. Add.	Invalid
		Write	L	H	L	H	H	H	L	X	Not Used
			L	L	H	H	H	H	L	Not Used	X

NOTES:
1. X = don't care.
2. For AT-BIOS compatibility, a host adapter decodes these I/O addresses. The flash drive decodes A0–2. The drive is in IDE-ATA mode if OE# is low at reset.
3. The host must decode addresses A11 and above to provide card enables CE1# and CE2# that place the drive on a 2-Kbyte boundary. The drive decodes A0–10.
4. The host must decode addresses A4 and above to provide card enables CE1# and CE2# that place the drive on a 16-byte boundary. The drive decodes A0–3.
5. The drive fully decodes these addresses using addresses A0–9.
6. In independent memory mode, each byte access to 000, 008, or even addresses 400–7FE presents a new data-buffer byte on D0–7, starting with the first buffer word's low byte. Each word access presents sequential data buffer words on D0–15. Byte access to 009 or odd addresses 401–7FF present high byte buffer data on D0–7 or D8–15, depending on CE1# and CE2#.

ADDRESSING MODES

IDE-ATA Mode

The host uses programmed I/O to address the drive. To access a desired register, the host asserts the drive's address lines (DA0–2), a drive select (CS1FX# or CS3FX#), and an I/O read or write strobe (DIOR# or DIOW#).

The host generates two interface drive selects. The high-order select, CS3FX#, accesses registers XX6 and XX7 (typically 3F6 and 3F7 or 376 and 377). The low order select, CS1FX#, accesses registers XX0–XX7 (typically 1F0–1F7 or 170–177).

IOCS16# tells the host that data register accesses activate data bus DD0-15. ECC transfers occur on DD0–7 only during read- or write-long operations.

In addition to the IDE-ATA mode, the flash drive supports four PCMCIA addressing modes summarized in Table 4.

Table 4. PCMCIA Addressing Modes

Conf. Index	Address Mode	Address	Description
0	Memory	000–00F, 400–7FF	Memory Mapped
1	I/O	0–F	I/O, 16 Contiguous Addresses
2	I/O	1F0–1F7, 3F6–3F7	Primary I/O
3	I/O	170–177, 376-377	Secondary I/O

PCMCIA-ATA Independent Memory Mode

Once the host configures the socket for PCMCIA-ATA independent memory mode, it uses memory instructions and a memory window, within the host's PCMCIA interface chip, to address the drive. To access a desired register, the host de-asserts REG#, asserts the drive's address lines (A0–10), card select(s) (CE1# and/or CE2#), and a read or write strobe (OE# or WE#).

The host generates two card selects. The high-order select, CS2#, accesses high-byte data or odd-register contents onto data lines D8–15. The low-order select, CS1#, accesses low-byte data or even-register contents onto data lines D0–7. Asserting both card selects while accessing the data register places high-byte register contents onto D8–15 and low-byte register contents onto D0–7. The host must decode addresses above A10 to place the drive on a 2-Kbyte memory boundary. The drive decodes addresses A0–10 as described in Table 3.

PCMCIA-ATA Independent I/O Mode

Once the host configures the socket for PCMCIA-ATA independent I/O mode, it addresses the drive using programmed I/O. To access a desired register, the host asserts REG#, the drive's address lines (A0-3), card select(s) (CE1# and/or CE2#), and an I/O read or write strobe (IORD# or IOWR#).

The host generates two card selects. The high-order select, CS2#, accesses high-byte data or odd-register contents onto data lines D8–15. The low-order select, CS1#, accesses low-byte data or even-register contents onto data lines D0–7. Asserting both card selects while accessing the data register places high-byte register contents onto D8–15 and low-byte register contents onto D0–7. The host must decode A4–10 to place the drive on a 16-byte I/O boundary. The drive decodes A0–3 as described in Table 3.

IOIS16# indicates that the selected address can transfer 16-bit data on D0–15. ECC transfers occur on D0–7 only during read- or write-long operations.

PCMCIA-ATA Primary or Secondary Mode (1F0–1F7, 3F6, 3F7 or 170–177, 376, 377)

Once the host configures the socket for PCMCIA-ATA primary or secondary drive-address I/O mode, it accesses the drive using programmed I/O. To access a desired register, the host asserts REG#, drive address lines A0–9, card select(s) CE1# and/or CE2#, and I/O read or write (IORD# or IOWR#).

The host generates two card selects. The high-order select, CS2#, accesses high-byte data or odd-register contents onto data lines D8–15. The low-order select, CS1#, accesses low-byte data or even-register contents onto data lines D0–7. Asserting both card selects while accessing the data register places high-byte data-register contents onto D8–15 and low-byte data-register contents onto D0–7. The host must decode addresses above A9 to generate drive selects when the primary (1F0–1F7, 3F6, and 3F7) or secondary (170–177, 376, and 377) drive-address range is selected. The drive decodes addresses A0–9 as described in Table 3.

IOIS16# indicates that the selected address can transfer 16-bit data on D0–15. ECC transfers occur on D0–7 only during read- or write-long operations.

Table 3 defines all I/O- and memory-mapped register addresses and their functions. ATA register descriptions follow.

PCMCIA FUNCTIONS

Attribute Memory Access

Attribute memory contains the card information structure (CIS). The host reads the CIS by asserting REG#-low, CE1#-low, CE2#-high, OE#-low, WE#-high and even addresses (A0 = VIL) starting at 0000h (see Table 3).

In addition to device, interface, features, and manufacturer information, the CIS tuples (Table 5) describe the four PCMCIA-ATA configurations described earlier.

The host picks one mode that fits its hardware and software requirements. It writes the mode's option index number to the Configuration Option register at card attribute address 200h.

Table 5. Card Information Structure Tuples

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
DEVICE INFORMATION TUPLE											
00	01	0	0	0	0	0	0	0	1	tpl_device	Common Memory Space
02	04	0	0	0	0	0	1	0	0	tpl_link	Link to Next Tuple
04	DF	1	1 Device Type Code	0	0	1 WPS	1	1 Dev Speed (see next)	1	device_ info	Dev Type = Function-Spec. Mem; Writable; Ext. Speed
06	4A	0 Ext	1	0 Mantisa 4.0 x	0	1	0	1 Speed Exp 100 ns	0	ext speed	No Add'l Speed Byte; 4.0 × 100 ns = 400 ns
08	01	0	0 No. of Addr Units − 1	0	0	0	0	0 Size	1	device_ size	# of Address Units = 1; Size Code = 2K (64K Max)
0A	FF	1	1	1	1	1	1	1	1	end	End Device Info
CONFIGURATION TUPLE											
0C	1A	0	0	0	1	1	0	1	0	tpl_config	Configuration Tuple Code
0E	05	0	0	0	0	0	1	0	1	tpl_link	
10	01	0 0 tpcc_rfsz		0	0 0 tpcc_rmsz		0	0 1 tpcc_rasz		tpcc_sz	tpcc_rmsk = tpcc_rmsz + 1 = 1 tpcc_radr = tpcc_rasz + 1 = 2
12	03	0 0 Reserved		0	0	0	0	1	1 Index Number of Final Card Config Table Entry	tpcc_last	Last Card Configuration Table Index Number
14	00	0	0	0	0	0	0	0	0	tpcc_radr	Configuration Register Base Address = 200h
16	02	0	0	0	0	0	0	1	0		
18	0F	0	0	0	0	1(1)	1(1)	1(1)	1	tpcc_rmsk	All Config Registers Present
CARD CONFIGURATION TABLE ENTRY TUPLE—Memory Space Definition											
1A	1B	0	0	0	1	1	0	1	1	cftbl_entry	Card Config Table Entry
1C	08	0	0	0	0	1	0	0	0	tpl_link	
1E	C0	1 I/F	1 Default	0	0 Configuration Entry Number		0	0	0	tpce_indx	I/F Byte Follows, Default, Config Table Index = 00
20	C0	1 Wait	1 Ry/By	0 WP	0 BVD	0	0 I/F Type		0	tpce_if	I/F-Def Byte; Wait Req'd, Ry/By Active, I/F = Memory
22	A1	1 misc	0 mem space	1	0 IRQ	0 I/O	0 Timing	0 V_{PP} & V_{CC}	1	tpce_fs	Feature-Sel; V_{CC} Described, Mem Space = 2-Bytes, Misc
24	01	0 resv'd	0 PD I	0 Peak I	0 Avg I	0 Stat I	0 Max V	0 Min V	1 Nom V	tpce_pd	Power-Description; One Field Described (Non V_{CC})
26	55	0 Ext	1	0 5.0 x	1	0	1	0 1V	1		V_{CC} Nom = 5V ±5%
28	08	0	0	0	0	1	0	0	0	tpce_ms	Memory Space Description; Map 8 × 256 = 2048 Bytes
2A	00	0	0	0	0	0	0	0	0		
2C	20	0 Ext	0 Resv'd	1 PwrDn	0 Rdonly	0 Audio	0 Max Twin Cards	0	0	tpce_mi	Miscellaneous Field; Power-Down Supported

Table 5. Card Information Structure Tuples (Continued)

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
CARD CONFIGURATION TABLE ENTRY TUPLE—I/O Space Definition											
2E	1B	0	0	0	1	1	0	1	1	cftbl_entry	Card Config Table Entry
30	0A	0	0	0	0	1	0	1	0	tpl_link	
32	C1	1 I/F	1 Default	0	0 Configuration Entry Number	0	0	0	1	tpce_indx	I/F Byte Follows, Default, Config Table Index = 01
34	41	0 Wait	1 Ry/By	0 WP	0 BVD	0	0 I/F type	0	1	tpce_if	I/F-Def; Ry/By Active, I/F = I/O
36	99	1 misc	0 mem space	0	1 IRQ	1 I/O	0 Timing	0 V_{PP} & V_{CC}	1	tpce_fs	Feature-Sel; V_{CC} Described, I/O, IRQ, Misc Present
38	01	0 resv'd	0 PD I	0 Peak I	0 Avg I	0 Static I	0 Max V	0 Min V	1 Nom V	tpce_pd	Power-Description; One Field Described (Non V_{CC})
3A	55	0 Ext	1	0 5.0 x	1	0	1	0 1V	1		V_{CC} Nom = 5V ±5%
3C	64	0 ext	1 16-Bit	1 8-Bit	0	0 I/O Address Lines	1	0	0	tpce_io	I/O Lines A0–3 Required, 8- or 16-Bit Bus
3E	F0	1 Share	1 Pulse	1 Level	1 Mask	0	0 IRQ Number = 0	0	0	tpce_ir	Interrupt Request; Share Level/Pulse IRQ, Mask
40	FF	1 IRQ7	1 IRQ6	1 IRQ5	1 IRQ4	1 IRQ3	1 IRQ2	1 IRQ1	1 IRQ0		IRQs 0–15 OK
42	FF	1 IRQ15	1 IRQ14	1 IRQ13	1 IRQ12	1 IRQ11	1 IRQ10	1 IRQ9	1 IRQ8		
44	20	0 Ext	0 Resv'd	1 PwrDn	0 Rd	0 Audio	0 Max Twin Cards	0	0	tpced_mi	Misc Field; Power-Down Mode Supported
CARD CONFIGURATION TABLE ENTRY TUPLE—Primary Drive-Address Definition (1F0–1F7, 3F6–3F7)											
46	1B	0	0	0	1	1	0	1	1	cftbl_entry	Card Config Table Entry
48	0C	0	0	0	0	1	1	0	0	tpl_link	
4A	82	1 I/F	0 Default	0	0 Configuration Entry Number	0	0	1	0	tpce_indx	I/F Byte Follows, not Default Config Table Index = 02
4C	41	0 Wait	1 Ry/By	0 WP	0 BVD	0	0 I/F type	0	1	tpce if	I/F-Def; Ry/By Active, I/F = I/O
4E	18	0 misc	0 mem space	0	1 IRQ	1 I/O	0 Timing	0 V_{PP} & V_{CC}	0	tpce fs	Feature-Select; I/O Space, IRQ Description Present
50	EA	1 Ext	1 16-bit	1 8-bit	0	1 I/O address lines	0	1	0	tpce io	I/O Lines A0–9 Required, 8- or 16-Bit Bus, Add Below

7

Table 5. Card Information Structure Tuples (Continued)

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
CARD CONFIGURATION TABLE ENTRY TUPLE— Primary Drive-Address Definition (1F0–1F7, 3F6–3F7) (Continued)											
52	61	0 length = 1 byte	1	1 add = 2 bytes	0	0	0	0	1 I/O addr ranges = 1 + 1 = 2		Number of I/O Address Range Fields that Follow
54 56	F0 01	1 0	1 0	1 0	1 0	0 0	0 0	0 0	0 1		I/O Start Address = 1F0h
58	07	0	0	0	0	0	1	1	1		I/O Block Length = 7 + 1 = 8
5A 5C	F6 03	1 0	1 0	1 0	1 0	0 0	1 0	1 1	0 1		I/O Start Address = 3F6h
5E	01	0	0	0	0	0	0	0	1		I/O Block Length = 1 + 1 = 2
60	EE	1 Share	1 Pulse	1 Level	0 Mask	1	1	1	0 IRQ Number	tpce ir	Interrupt Request; Share, Level/Pulse IRQ = 14
CARD CONFIGURATION TABLE ENTRY TUPLE— Secondary Drive-Address Definition (170–177, 376–377)											
62	1B	0	0	0	1	1	0	1	1	cftbl_entry	Card Config Table Entry
64	0C	0	0	0	0	1	1	0	0	tpl_link	
66	83	1 I/F	0 Default	0	0	0	0	1	1 Configuration Entry Number	tpce_indx	I/F Byte Follows, not Default Config Table Index = 03
68	41	0 Wait	1 Ry/By	0 WP	0 BVD	0	0	0	1 I/F type	tpce if	I/F-Def; Ry/By Active, I/F = I/O
6A	18	0 misc	0	0 mem space	1 IRQ	1 I/O	0 Timing	0	0 V_{PP} & V_{CC}	tpce fs	Feature-Select; I/O Spsce, IRQ Description Present
6C	EA	1 Ext	1 16-bit	1 8-bit	0	1	0	1	0 I/O address lines	tpce_io	I/O Lines A0–9 Required, 8- or 16-Bit Bus, Add Below
6E	61	0 length = 1 byte	1	1 add = 2bytes	0	0	0	0	1 I/O addr ranges = 1 + 1 = 2		Number of I/O Address Range fields that Follow
70 72	70 01	0 0	1 0	1 0	1 0	0 0	0 0	0 0	0 1		I/O Start Address = 170h
74	07	0	0	0	0	0	1	1	1		I/O Block Length = 7 + 1 = 8
76 78	76 03	0 0	1 0	1 0	1 0	0 0	1 0	1 1	0 1		I/O Start Address = 376h
7A	01	0	0	0	0	0	0	0	1		I/O Block Length = 1 + 1 = 2
7C	EE	1 Share	1 Pulse	1 Level	0 Mask	1	1	1	0 IRQ Number	tpce ir	Interrupt Request; Share, Level/Pulse IRQ = 14

Table 5. Card Information Structure Tuples (Continued)

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
FUNCTION ID TUPLE FOR DISK FUNCTION											
7E	21	0	0	1	0	0	0	0	1	tpl_funce	Disk Drive Function ID Tuple
80	02	0	0	0	0	0	0	1	0	tpl_link	
82	04	0	0	0	0	0	1	0	0	tplfid_func	04 = Fixed Disk Function
84	01	0	0	0	0	0	0	0	1	tplfid_ sysinit	System Init Flags; 01 = Host POST Must Configure Card
DISK DEVICE INTERFACE FUNCTION EXTENSION TUPLE											
86	22	0	0	1	0	0	0	1	0	tpl_funce	Disk Function Extension
88	02	0	0	0	0	0	0	1	0	tpl_link	
8A	01	0	0	0	0	0	0	0	1	tplfe_type	Disk I/F Extension Type
8C	01	0	0	0	0	0	0	0	1	tplfd_data	PC Card ATA I/F Type
PC CARD ATA FEATURES FUNCTION EXTENSION TUPLE											
8E	22	0	0	1	0	0	0	1	0	tpl_funce	Disk Function Extension
90	03	0	0	0	0	0	0	1	1	tpl_link	
92	02	0	0	0	0	0	0	1	0	tplfe_type	Disk I/F Extension Type
94	0C	0 Reserved (must be 0's)				1 drv ID	1 Silic'n	0 V_{PP} Req'd	0	tplfe_data	Card ATA Feature Byte 1; Rotating Dev, Model/SN ID
96	0F	0 Resv'd	0 I/O16	0 Index	0 Floppy	1 autopwr	1 Idle	1 stndby	1 Sleep	tplfe_data	Csard ATA Feature Byte 2; Supports Low-Power Modes
JEDEC PROGRAMMING INFORMATION TUPLE FOR COMMON MEMORY											
98	18	0	0	0	1	1	0	0	0	tpl_jed_c	ATA JEDEC ID
9A	02	0	0	0	0	0	0	1	0	tpl_link	
9C	DF	1	1	0	1	1	1	1	1	JedecID 1	Manufacturer ID = DF
9E	01	0	0	0	0	0	0	0	1	JedecID 2	Dev ID = 1 for ATA, No V_{PP}
MANUFACTURER IDENTIFICATION TUPLE											
A0	20	0	0	1	0	0	0	0	0	tpl_manfid	Manufacturers ID Tuple
A2	04	0	0	0	0	0	1	0	0	tpl_link	
A4 A6	89 00	1 0	0 0	0 0	0 0	1 0	0 0	0 0	1 0	tplmid_ manf	Manufacturer Code = 89h
A8 AA	AF 00	1 0	0 0	1 0	0 0	1 0	1 0	1 0	1 0	tplmif_ card	Part Number = AFh

7

Table 5. Card Information Structure Tuples (Continued)

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
LEVEL 1 VERSION/PRODUCT INFORMATION TUPLE											
AC	15	0	0	0	1	0	1	0	1	tpl_vers_1	Version 1 Compliance
AE	1C	0	0	0	1	1	1	0	1	tpl_link	
B0	04	0	0	0	0	0	1	0	0	tpllv1_maj	Major Version Number=04
B2	01	0	0	0	0	0	0	0	1	tpllv1_min	Miinor Version Number=01
B4	49	0	1	0	0	1	0	0	1	I	tpllv1_info
B6	6E	0	1	1	0	1	1	1	0	n	Manufacturer's Information
B8	74	0	1	1	1	0	1	0	0	t	
BA	65	0	1	1	0	0	1	0	1	e	
BC	6C	0	1	1	0	1	1	0	0	l	
BE	00	0	0	0	0	0	0	0	0		End Maufacturer Info
C0	46	0	1	0	0	0	1	1	0	F	Product Information String
C2	6C	0	1	1	0	1	1	0	0	l	
C4	61	0	1	1	0	0	0	0	1	a	
C6	73	0	1	1	1	0	0	1	1	s	
C8	68	0	1	1	0	1	0	0	0	h	
CA	44	0	1	0	0	0	1	0	0	D	
CC	72	0	1	1	1	0	0	1	0	r	
CE	69	0	1	1	0	1	0	0	1	i	
D0	76	0	1	1	1	0	1	1	0	v	
D2	65	0	1	1	0	0	1	0	1	e	
D4	00	0	0	0	0	0	0	0	0		End of Product Info String
D6	69	0	1	1	0	1	0	0	1	i	Additional Product Info
D8	46	0	1	0	0	0	1	1	0	F	
DA	44	0	1	0	0	0	1	0	0	D	
DC	50	0	1	0	1	0	0	0	0	P	
DE	32	0	0	1	1	0	0	1	0	2	
E0	53	0	1	0	1	0	0	1	1	S	
E2	41	0	1	0	0	0	0	0	1	A	
E4	00	0	0	0	0	0	0	0	0		End Additional Product Info
E6	FF	1	1	1	1	1	1	1	1		End Device Info
NO-LINK CONTROL TUPLE (End of Tuples Marker)											
E8	14	0	0	0	1	0	1	0	0	tpl_no_lnk	No-Link Control Tuple
EA	00	0	0	0	0	0	0	0	0	tpl_link	
EC	FF	1	1	1	1	1	1	1	1	tpl_end	End of the Tuple Chain

NOTE:
1. Engineering samples (marked ES) and early production units (marked iFD0xxP2SA B) do not support the Card Configuration and Status, Pin Replacement, and Socket and Copy registers.

PCMCIA Card Configuration Registers

These read/write one-byte registers are located on even-byte attribute-plane addresses to ensure single-cycle access in both 8- and 16-bit systems. Available card status information allows arbitration between resources that share interrupts and memory-only-card status of pins 16, 33, 62, and 63. Table 6 describes these registers.

CONFIGURATION OPTION REGISTER

The host uses the read/write Configuration Option register to configure the drive for one of its four PCMCIA-ATA addressing modes, establish the interrupt signal mode, and issue a soft reset.

7	6	5	4	3	2	1	0
SRST	IRQLvl	Configuration Index					

SRST — Resets the card when 1. When 0 (default), the card is unconfigured, similar to the state following hardware reset or power-on (which also reset this bit to 0).

IRQLvl — Selects level mode interrupts when 1; pulse mode interrupts when 0 (default).

Conf IDX — The host chooses an option from the card's configuration table tuples and writes that option's Configuration Index number into this field. When zero (default), the memory-only interface is chosen; I/O accesses are disabled.

CARD CONFIGURATION AND STATUS REGISTER

The Card Configuration and Status register contains card condition information.

7	6	5	4	3	2	1	0
Chng	SigChg	IOis8	0	Audio	PwrDn	Intr	0

Chng — The Change bit indicates that a Pin Replacement register bit was changed.

SigChg — The host sets/resets the Signal Changed bit to enable/disable a state-change signal from the status register. When set and the drive is configured for I/O, Chng controls pin 63 and is called the Changed Status signal. This bit should be 0 (BVD1/STSCHG# held high when configured for I/O) if no state change signal is desired.

IOis8 — Must be set if the host can only perform 8-bit I/O accesses.

Audio — Audio is not supported.

PwrDn — Setting PwrDn places the drive in sleep mode. Host-initiated ATA task-file-register commands can also invoke low-power modes.

Intr — This bit represents the interrupt request's internal state. Its value is available whether or not interrupts are configured. It remains true until the initiating-interrupt request is serviced.

Table 6. PCMCIA Card Configuration Registers Truth Table

Register	Add. Offset(1)	R/W	REG#	CE2#	CE1#	OE#	WE#	D15–D8	D7–D0
Configuration Option	200h	Read	L	H	L	L	H	Invalid	Option
		Write	L	H	L	H	L	Invalid	Option
Card Configuration and Status	202h(2)	Read	L	H	L	L	H	Invalid	Status
		Write	L	H	L	H	L	Invalid	Config.
Pin Replacement	204h(2)	Read	L	H	L	L	H	Invalid	Pin Status
		Write	L	H	L	H	L	Invalid	Pin Status
Socket and Copy	206h(2)	Read	L	H	L	L	H	Invalid	Socket ID
		Write	L	H	L	H	L	Invalid	Socket ID

NOTES:
1. The host obtains the attribute-memory address offset from the Configuration Tuple's TPCC_RADR field when it reads the drive's CIS.
2. Engineering samples (marked ES) and early production units (marked iFD0xxP2SA B) do not support these registers.

PIN REPLACEMENT REGISTER

The Pin Replacement register provides card status information that is otherwise provided on memory-only interface pins 16, 33, 62, and 63. When written, bits 0–3 are masks for setting corresponding bits 4–7.

7	6	5	4	3	2	1	0
CBvd 1	CBvd 2	Crdy/ bsy#	CWP	RBvd 1	RBvd 2	Rrdy/ bsy#	RWP

CBvd1,2 Set when written, otherwise zero.

Crdy/ bsy# Set when Rrdy/bsy# changes state or when written by the host.

CWP Zero unless set by the host.

RBvd1,2 Set unless cleared by the host.

Rrdy/ bsy# Internal ready/busy# state when I/O mode uses RDY/BSY# pin for interrupt.

RWP Zero, since the flash drive has no write protect switch, unless set by the host.

SOCKET AND COPY REGISTER

The Socket and Copy register allows the drive to distinguish between similar drives at the same address. The flash drive does not support this feature.

7	6	5	4	3	2	1	0
0	Copy Number			Socket Number			

Copy # The twin-card option is not supported.

Socket # The socket number is ignored.

ATA FUNCTIONS

AT-Attachment Compatibility

The flash drive is hardware and software compatible with the IDE-AT-Attachment interface standard. It supports all mandatory, most optional, and several vendor unique commands.

The drive can operate in either CHS (cylinder, head, sector) or LBA (logical block address) mode on a by-command basis. *Identify drive* command parameters indicate LBA mode compatibility.

If the host selects LBA mode via the Drive/Head register, Drive/Head bits 3–0, Sector Number, and Cylinder-low and high registers contain the zero-based LBA. In LBA mode, drive-sectors are linearly mapped with LBA0 at:

cylinder = 0, head = 0, sector = 1.

Regardless of host-set translation mode geometry, a given sector's LBA address does not change:

LBA = [(cylinder * number-of-heads + heads) * sectors/track] + sector − 1.

Register Description

In the following, unused read bits are "don't cares" and unused write bits are zeros. The data register is 16 bits wide; all others are 8 bits wide.

DATA REGISTER (READ/WRITE)

Data block PIO (programmed I/O) transfers between the drive's data buffer and the host use this 16-bit register. A *format track* command transfers sector information through this register.

F	E	D	C	B	A	9	8
D15	D14	D13	D12	D11	D10	D9	D8
D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

ERROR REGISTER (READ)

The Error register contains last drive-command-executed status or a diagnostic code. At any command completion, except *execute drive diagnostic*, the Error register's contents are valid when the Status register's ERR is 1. It contains a diagnostic code (see Table 8) following power-on, reset, or *execute drive diagnostic* command.

7	6	5	4	3	2	1	0
BBK	UNC	MC	IDNF	MCR	ABRT	T0NF	AMNF

BBK Bad Block mark detected in the requested sector's ID field.

UNC UNCorrectable data error encountered.

MC Removable Media Changed; media access ability has changed—not supported.

IDNF Requested sector's ID-field Not Found.

MCR Media Change Request indicates that the removable-media drive's latch has changed, indicating that the user wishes to remove the media—not supported.

ABRT Drive status error or aborted invalid command.

T0NF Track 0 Not Found during a recalibrate command.

AMNF Address Mark Not Found after finding the correct ID field.

Unused bits are zero.

SET FEATURE REGISTER (WRITE)

This register enables drive-specific features. See the *set features* or *Intel get/set features* command descriptions.

SECTOR COUNT REGISTER (READ/WRITE)

This register contains the number of data sectors to be transferred during a read or write operation. A zero register value specifies 256 sectors. The command was successful if this register is zero at command completion. If the request is not completed, the register contains the number of sectors left to be transferred.

Some commands (e.g. *initialize drive parameters* or *format track*) may redefine this register's contents.

7	6	5	4	3	2	1	0
SC7	SC6	SC5	SC4	SC3	SC2	SC1	SC0

SECTOR NUMBER REGISTER (READ/WRITE)

In CHS (cylinder, head, sector) mode, this register contains the subsequent command's starting sector number, which can be from 1 to the maximum number of sectors per track. See the command descriptions for register contents at command completion (whether successful or unsuccessful).

In LBA (logical block address) mode, this register contains LBA bits 0–7. It reflects updated LBA bits 0–7 at command completion.

7	6	5	4	3	2	1	0
SN7	SN6	SN5	SN4	SN3	SN2	SN1	SN0

CYLINDER LOW REGISTER (READ/WRITE)

In CHS mode, this register contains the current cylinder number's low-order 8 bits at any disk access start or at command completion.

In LBA mode, this register contains LBA bits 8–15 and reflects their status at command completion.

7	6	5	4	3	2	1	0
CL7	CL6	CL5	CL4	CL3	CL2	CL1	CL0

CYLINDER HIGH REGISTER (READ/WRITE)

In CHS mode, this register contains the current cylinder number's high-order 8 bits at any disk access start or at command completion.

In LBA mode, this register contains LBA bits 16–23 and reflects their status at command completion.

7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

DRIVE/HEAD REGISTER (READ/WRITE)

This register contains the drive and head numbers (heads minus 1, when executing an *initialize drive parameters* command).

7	6	5	4	3	2	1	0
1	L	1	DRV	HS3	HS2	HS1	HS0

L Address mode select. 0=CHS (cylinder, head, sector) mode; 1=LBA (logical block address) mode.

DRV Binary drive select number; DRV=0 selects drive 0, DRV=1 is ignored (on HDDs this bit selects drive 1).

HS3–0 If L=0: HS3–0 is the selected head's binary coded address (e.g. if HS3–0 is 0011b, head 3 is selected). At command completion, these bits reflect the currently selected head.

If L=1: HS3–0 contain LBA bits 24–27 and reflect their status at command completion.

STATUS REGISTER (READ)

This register contains drive status, which is updated at each command's completion. When BSY is cleared, register bits are valid within 400 ns. If set, no other Status register bits are valid. A host read,

while interrupt is pending, constitutes interrupt acknowledge which clears any pending interrupt.

7	6	5	4	3	2	1	0
BSY	DRDY	DWF	DSC	DRQ	CORR	IDX	ERR

BSY The drive can access command block registers (the host should not access them) when set; any command block register read returns the Status register contents. The drive sets BSY:

within 400 ns after RESET# negation or after setting the Device Control register's SRST. Following reset, the drive sets BSY for up to 31 seconds.

within 400 ns of host command: *read, read long, read buffer, seek, recalibrate, read verify, initialize drive parameters, identify drive, execute drive diagnostic.*

within 5 µs after a 512 byte *write, format track,* or *write buffer* command or data-plus-ECC *write long* command.

DRDY Drive Ready indicates that the drive can respond to a command. On error, DRDY changes only after the host reads the Status register; the bit again indicates drive readiness. Power-on clears this bit; it remains clear until the drive can accept a command.

DWF Drive Write Fault status. On error, it changes only after the host reads the Status register; the bit again indicates the current write fault status.

DSC Drive Seek Complete indicates that the requested sector was found. On error, it changes only after a host Status register read; the bit again indicates the current seek complete status.

DRQ Data Request indicates that the drive can transfer a data word or byte.

CORR Indicates that a data error was corrected; transfer is not terminated.

IDX Index is not used.

ERR An Error occurred during the previous command's execution. The Error register indicates the error's cause.

COMMAND REGISTER (WRITE)

Execution commences when this register receives a host command. Table 7 lists executable command codes and indicates valid parameters.

ALTERNATE STATUS REGISTER (READ)

This register contains command block Status information (see Status register). Unlike the Status register, reading this register does not acknowledge or clear an interrupt.

7	6	5	4	3	2	1	0
BSY	DRDY	DWF	DSC	DRQ	CORR	IDX	ERR

DRIVE CONTROL REGISTER (WRITE)

This register contains two control bits:

7	6	5	4	3	2	1	0
—	—	—	—	—	SRST	IEN#	—

SRST Resets the drive when set, enables it when cleared.

IEN# Controls Interrupt Enable to the host (three-state pin INTRQ) when the drive is selected. When inactive (1) or drive not selected, INTRQ is high-Z regardless of a pending interrupt.

DRIVE ADDRESS REGISTER (READ)

This register reflects the drive and its heads.

7	6	5	4	3	2	1	0
High-Z	WTG#	HS3#	HS2#	HS1#	HS0#	DS1#	DS0#

High-Z For floppy drive address-space compatibility, bit 7 is not driven (it is high-Z) when this register's address is accessed in IDE-ATA mode. This allows a floppy-drive's disk-change bit to provide bit 7 data.

WTG# Write Gate bit is active during in-progress disk writes (not supported).

HS3–0# Ones complement of the currently selected head number. For example, head 3 is selected if HS3–0 is 1100b.

DS1# Drive Select 1 is low when drive 1 is selected and active (not supported).

DS0# Drive Select 0 is low when drive 0 is selected and active.

ATA Command Descriptions

Writing Command register codes and block-register parameters issues commands. The drive responds and executes a new command by aborting an in-progress command, even if it can be completed.

Table 7. Command Register Codes

Command Code (hex)	Command Name	Supported	FR	SC	SN	CY	DH
10–1F	Recalibrate	✓					D
20	Read Sector(s) (w/ retry)	✓		y	y	y	y
21	Read Sector(s) (w/o retry)	✓		y	y	y	y
22	Read Long (w/ retry)	✓		y	y	y	y
23	Read Long (w/o retry)	✓		y	y	y	y
30	Write Sector(s) (w/ retry)	✓	*	y	y	y	y
31	Write Sector(s) (w/o retry)	✓	*	y	y	y	y
32	Write Long (w/ retry)	✓	*	y	y	y	y
33	Write Long (w/o retry)	✓	*	y	y	y	y
40	Read Verify Sector(s) (w/ retry)	✓		y	y	y	y
41	Read Verify Sector(s) (w/o retry)	✓		y	y	y	y
50	Format Track	✓	*	y		y	y
70–7F	Seek	✓			y	y	y
90	Execute Drive Diagnostic	✓					D
91	Initialize Drive Parameters	✓		y			y
94 E0	Standby Immediate	✓					D
95 E1	Idle Immediate	✓					D
96 E2	Standby	✓		y			D
97 E3	Idle	✓		y			D
98 E5	Check Power Mode	✓		y			D
99 E6	Set Sleep Mode	✓					D
9A	Intel Get/Set Features	✓		y	y	y	y
C4	Read Multiple	✓		y	y	y	y
C5	Write Multiple	✓	*	y	y	y	y
C6	Set Multiple Mode	✓		y			D
E4	Read Buffer	✓					D
E8	Write Buffer	✓					D
EC	Identify Drive	✓					D
EF	Set Features	✓	y				D
80–8F, C0–C3, F0–FF	Vendor Unique	✓	y	y	y	y	y

CY = Cylinder Registers
DH = Drive/Head Register
SC = Sector Count Register
SN = Sector Number Register
FR = Features Register (see command descriptions)
y = Register has a valid parameter.
D = Only the drive, not the head, parameter is valid.
* = Maintained for compatibility

7

RECALIBRATE, 1X

Recalibrate completes and returns status.

READ SECTOR(S), 20 OR 21

This command reads from 1 to 256 sectors as specified in the Sector Count register; a sector count of 0 requests 256 sectors. The Sector Number register specifies the beginning sector.

DRQ is always set prior to data transfer, even in the presence of an error condition.

If an error occurs, read terminates at the erred sector. The sector buffer retains flawed data.

At command completion, command block registers contain the last-read sector's cylinder, head, and sector numbers (or LBA).

READ LONG, 22 OR 23

The *read long* command is similar to the *read sectors* command but returns the desired sector's data field and ECC bytes (8 bits wide). During a *read long* command, the drive does not check the ECC bytes to determine data errors. Only single-sector read long operations are supported.

WRITE SECTOR(S), 30 OR 31

This command writes from 1 to 256 sectors as specified in the Sector Count register beginning at the specified sector; a sector count of 0 transfers 256 sectors. The Sector Number register specifies the beginning sector. Buffer data followed by ECC bytes are written to the sector's data field.

If an error occurs during a multiple sector write, writing terminates at the erred sector. The host may read the command block to discern the error and erred sector. See the Error Posting section for Error and Status register contents.

At command completion, command block registers contain the last written sector's cylinder, head, and sector number (or LBA).

WRITE LONG, 32 OR 33

This command is similar to the *write sectors* command but writes data and ECC bytes directly from the sector buffer; the drive does not generate ECC bytes itself. Only single sector write long operations are supported.

READ VERIFY SECTOR(S), 40 OR 41

This command, identical to the *read sectors* command, transfers no data and resets DRQ. When the requested sectors are verified, the drive clears BSY and generates an interrupt.

If an error occurs, verify terminates at the erred sector. The Sector Count register contains the number of unverified sectors.

Upon command completion, command block registers contain the last verified sector's cylinder, head, and sector number (or LBA).

FORMAT TRACK, 50

This command marks a track's sectors as "deleted." The next time a reclamation procedure is invoked, these sectors are erased and returned to the available flash-memory pool.

Cylinder High and Cylinder Low registers specify the track address (or LBA). The Sector Count register specifies the number of sectors. When the command is accepted, the drive sets DRQ and waits for the host to fill the sector buffer. The sector buffer's contents are not written to the media and are, in fact, ignored. The drive then clears DRQ, sets BSY, and begins command execution.

SEEK, 7X

This command completes and immediately returns successful seek status.

EXECUTE DRIVE DIAGNOSTIC, 90

This command performs internal self-diagnostic tests. The drive posts its diagnostic results, clears BSY, generates an interrupt, and loads its Error register's status code. Table 8 shows 8-bit Error register diagnostic codes.

Table 8. Diagnostic Codes

Code	Description
01h	No Error Detected
02h	Formatter Device Error
03h	Sector Buffer Error
04h	ECC Circuitry Error
05h	Controlling Microprocessor Error
8xh	Drive 1 Failed (not used)

INITIALIZE DRIVE PARAMETERS, 91

This command sets the sectors per track (Sector Count register) and the number of heads (minus 1) per cylinder (Drive/Head register). Upon command receipt, the drive sets BSY, saves the parameters, clears BSY, and generates an interrupt.

This command does not check sector-count and head-value validity. If invalid, no error is posted until another command makes an illegal access.

STANDBY IMMEDIATE, 94 OR E0

This command places the drive in standby mode. The drive sets the interrupt before completing the standby mode transition. If the drive is already in standby, the standby sequence is not re-executed.

IDLE IMMEDIATE, 95 OR E1

At this command, the drive sets BSY, enters idle mode, clears BSY, and generates an interrupt (even if it has not fully transitioned to idle mode).

STANDBY, 96 OR E2

This command places the drive in standby mode. The drive sets the interrupt before transitioning to standby mode. If the drive is already in standby mode, the standby sequence is not re-executed.

IDLE, 97 OR E3

Idle causes the drive to set BSY, enter idle mode, clear BSY, and generate an interrupt (even if the drive has not fully transitioned to idle mode). The spinup sequence is not executed if the drive is already active.

CHECK POWER MODE, 98 OR E5

This command checks the power mode. If the drive is in, going to, or recovering from standby mode, it:

- sets BSY
- clears the Sector Count register
- clears BSY
- generates an interrupt.

If it is in idle mode, the drive:

- sets BSY
- sets the Sector Count register to FFh
- clears BSY
- generates an interrupt.

SET SLEEP MODE, 99 OR E6

This command places the drive in sleep mode. The drive powers down all internal circuitry, clears BSY, and generates an interrupt. If the drive is already asleep, the sleep sequence is re-executed.

A drive self-powers on following a command register write, power-on, hardware reset, or host-issued software reset.

INTEL GET/SET FEATURES, 9A

The *Intel get/set features* command, with a sector count register subfunction code, gets or sets special features. A 0 subfunction code retrieves features, and a 1 subfunction code sets features. Table 9 describes the Intel unique features.

READ MULTIPLE, C4

The *read multiple* command is similar to the *read sectors* command. An interrupt is not generated by every sector but by a block transfer. The *set multiple mode* command, which must be executed prior to the *read multiple* command, determines the sector block count to be transferred without intervening interrupts. DRQ qualification is required only at the data block's start, not at each sector.

The Sector Count register contains the number of sectors requested (not the block count). If a non-even block count is requested, all possible full blocks are transferred followed by a partial block; remainder of sector count/block count.

The *read multiple* command is rejected with an aborted-command error if it is executed before a *set multiple mode* command or if *read multiple* commands are disabled.

Disk errors encountered during *read multiple* commands are posted at the beginning of a block or partial block transfer. DRQ is still set and the data, including corrupted-data, transfers normally.

Command block register contents are undefined following a sector-in-error data block transfer. The host should retry the transfer as individual requests to obtain error information.

Subsequent or partial blocks are transferred only if the data-error was correctable. Other errors stop the command after transferring the error-containing block. Interrupts are generated when the beginning of each block or partial block transfer sets DRQ.

WRITE MULTIPLE, C5

This command is similar to the *write sectors* command. The drive sets BSY within 400 ns after the command. Interrupts don't occur after each sector transfer but after a block transfer. The *set multiple* command executed prior to the *write multiple* command defines the number of sectors transferred without intervening interrupts. DRQ transfer-qualification is required only at the data block start, not at each sector.

The Sector Count register contains the number of sectors requested (not the block count). If a non-even block count is requested, all possible full blocks are transferred followed by a partial block; remainder of sector count/block count.

The *write multiple* command is rejected with an aborted-command error if it is executed before a *set multiple mode* command or if *write multiple* commands are disabled.

Disk errors during *write multiple* commands are posted after the attempted block or partial block disk write. The *write multiple* command ends with the erred sector, even if it was in the middle of a block; subsequent blocks are not transferred. Setting DRQ generates interrupts at the beginning of each block or partial block.

Command block register contents are undefined following a sector-in-error data block transfer. The host should retry the transfer as individual requests to obtain error information.

SET MULTIPLE MODE, C6

This command lets the drive perform read- and write-multiple operations and establishes the command block count. At the command, the drive sets BSY and checks the Sector Count register.

The Sector Count register receives the sectors-per-block number. The flash drive supports 2, 4, 8, and 16 sector block sizes.

If the Sector Count register value is valid and the block count is supported, command execution is enabled and subsequent *read-* and *write-multiple* commands use this value. If a block count is not supported or the Sector Count register contains 0, an aborted-command error is posted, and *read-* and *write-multiple* commands are disabled.

At power-on, or after hardware reset, *read-* and *write-multiple* modes are disabled.

READ DMA, C8 OR C9

This command is not supported.

WRITE DMA, CA OR CB

This command is not supported.

READ BUFFER, E4

This command allows the host to read sector buffer contents. When issued, the drive sets BSY, sets the sector buffer for a read operation, sets DRQ, clears BSY, and generates an interrupt. The host then reads up to 512 data bytes.

The drive synchronizes *write* and *read buffer* commands so sequential commands access the same 512 buffer bytes.

WRITE BUFFER, E8

This command allows the host to overwrite any data pattern into the drive's sector buffer. The drive synchronizes sequential *write* and *read buffer* commands so they access the same buffer bytes.

IDENTIFY DRIVE, EC

The *identify drive* command allows the host to receive drive parameter information. When issued, the drive sets BSY, transfers parameter information to the sector buffer, sets DRQ, and generates an interrupt. The host then reads the sector buffer. Table 10 defines arguments and their meanings. Reserved bits or words are zero.

SET FEATURES, EF

The host sets features by writing the following values to the Set Feature register and then writing EF to the command register.

55 Disable read look-ahead
AA Enable read look-ahead (not supported)
81 Disable 8-bit data transfers (not supported)
82 Disable write cache (not supported)
CC Enable power-on defaults

Error Posting

Table 11 defines each command's valid errors. See Error and Status register bit definitions.

Table 9. Intel Vendor Unique Arguments (Command Register = 9A)

Subfunction	Set Features	Sector No.	Sector Count	Cylinder	Drive/Head
Get Features	Don't Care	9A	0	Vendor Word	Don't Care
Set Features	Don't Care	9A	1	Vendor Word	Don't Care
ATA Code Update	Sectors Count	9A	5	Security ID = FEED	A1
Low-Level Format	Don't Care	9A	6	Don't Care	Don't Care

NOTE:
Cylinder register low = vendor argument low-byte; cylinder register high = vendor argument high-byte.

INTEL VENDOR WORD

F	E	D	C	B	A	9	8
Compression	ECC	Auto Sleep	Auto Pwr-dn	Reserved	Power mode		Master/slave
Reserved	Reserved	Reset Mode	Reserved	Cache	Command Governing		Drive C
7	6	5	4	3	2	1	0

Bit 0 Drive is Master (drive C)

Bits 1–2 Max. governing during clean-up:

Bit 2	Bit 1	Cleanup Time (seconds)
0	0	2
0	1	4
1	0	4
1	1	8

Bit 3 Enable Caching
Bit 4 Reserved
Bit 5 Enable special reset mode (1)
Bits 6–7 Reserved
Bit 8 Master/slave; Non-Conner = 0, Conner = 1

Bits 9–A Clean-up state machine power modes

Bit A	Bit 9	Total CSMs	Read CSMs	Write CSMs	Perf. Rating	Power Rating
0	0	1	0	0	Lowest	Lowest
0	1	1	1	0		
1	0	2	1	1		
1	1	2	2	1	Highest	Highest

Bit B Reserved
Bit C Auto power-down; 0 = enabled; 1 = disabled
Bit D Auto standby = 0, auto sleep = 1 (default)
Bit E ECC enabled (0) or disabled (1)
Bit F compression; enabled = 0, disabled = 1 (not supported).

7

Table 10. Identify Drive Arguments

Word[1]	Bit	Value 5M/10M	Description
0		848C	General configuration bit-significant information:
	15		1 = non-magnetic drive
	14		0 = format speed tolerance gap not required
	13		0 = track offset option available
	12		0 = data strobe offset option available
	11		0 = rotational speed tolerance is > 0.5%
	10		1 = disk transfer rate > 10 MB/s
	9		0 = disk transfer rate > 5 MB/s but ≤ 10 MB/s
	8		0 = disk transfer rate ≤ 5 MB/s
	7		1 = removable cartridge drive
	6		0 = fixed drive
	5		0 = spindle motor control option implemented
	4		0 = head switch time > 15 μs
	3		1 = not MFM encoded
	2		1 = soft sectored
	1		0 = hard sectored
	0		0 = reserved
1[2]		009C/0140	Number of cylinders in the default translation mode
3[2]		0002	Number of heads in the default translation mode
4[2]		4000	Number of unformatted bytes per track in the default translation mode
5[2]		0200	Number of unformatted bytes per sector in the default translation mode
6[2]		0020	Number of sectors per track in the default translation mode
10–19		Unique	Serial No. 20 ASCII chars, right justified, space padded (20h); 0000h = none
20[3]		0003	Buffer Type 0000h Not specified; 0004h–FFFFh Reserved 0001h Single port single sector buffer; no simultaneous host-disk transfer 0002h Dual port multi-sector buffer; simultaneous host-disk transfers 0003h Dual port multi-sector buffer; simultaneous transfers with read
21		0040	Buffer size in 512-byte increments (0000h = not specified)
22		0004	Number of ECC bytes passed on *read/write long* commands (0000h = none)
23–26		Unique	Firmware rev.; 8 ASCII chars, left justified, space padded (20h); 0000h = none
27–46[4]			Model No. 40 ASCII chars, left justified, space padded (20h); 0000h = none
47	7–0	8008	Max *read/write multiple* sectors transferred/interrupt (00h = not allowed)
48		0000	0000h = cannot perform doubleword I/O; 0001h = can perform doubleword I/O
49	15–10	0201	Capabilities; 0 = reserved
	9		1 = LBA mode supported; 0 = CHS mode supported
	8		1 = DMA supported; 0 = DMA not supported
51[5]	15–8	0100	PIO data transfer cycle timing mode
52	15–8	0100	DMA data transfer cycle timing mode (not supported)
54		009C/0140	Number of current cylinders in the current translation mode
55		0002	Number of current heads in the current translation mode
56		0020	Number of current sectors-per-track in the current translation mode
57–58		2700/5000	Sector capacity, excluding device-specific sectors, = words (54 * 55 * 56)
59	8	0000	1 = Multiple sector setting, bits 7–0, is valid
	7–0		No. of sectors set to transfer per interrupt on *read/write multiple* commands

Table 10. Identify Drive Arguments (Continued)

Word(1)	Bit	Value 5M/10M	Description
60–61		26FF/4FFF	Total user addressable sectors if LBA mode is supported, CHS mode = 0
62	15–8	0203	Single-word DMA transfer mode active, indicated by bits 7–0; (not supported)
	7–0		Single-word DMA transfer modes available; (not supported)
63	15–8	0203	Multi-word DMA transfer mode active, indicated by bits 7–0; (not supported)
	7–0		Multi-word DMA transfer modes available; (not supported)

NOTES:
1. Words not listed are ATA or Intel reserved.
2. Initialize drive parameters does not affect words 1–6.
3. These codes, not typically used by an operating system, are for diagnostic program initialization.
4. 5 MB drive = iFD005P2SA, 10 MB drive = iFD010P2SA.
5. Each ATA device's transfer timing falls into unique parametric timing specification categories, determined by comparing the cycle time specified in Figure 7 with bits 15–8. The flash drive uses mode 1 which transfers 5.22 MB/s.

Table 11. Register Contents during Error Posting

Command	Error Register						Status Register				
	BBK	UNC	IDNF	ABRT	TK0NF	AMNF	DRDY	DWF	DSC	CORR	ERR
Check Power Mode				V			V	V	V		V
Execute Drive Diags											V
Format Track			V	V			V	V	V		V
Identify Drive				V			V	V	V		V
Idle				V			V	V	V		V
Idle Immediate				V			V	V	V		V
Initialize Drive Params							V	V	V		
Recalibrate				V	V		V	V	V		V
Read Buffer				V			V	V	V		V
Read Long	V	V	V	V		V	V	V	V	V	V
Read Multiple	V	V	V	V		V	V	V	V	V	V
Read Sector(s)	V	V	V	V		V	V	V	V	V	V
Read Verify Sector(s)	V	V	V	V		V	V	V	V	V	V
Seek			V	V			V	V	V		V
Set Features				V			V	V	V		V
Set Multiple Mode				V			V	V	V		V
Sleep				V			V	V	V		V
Standby				V			V	V	V		V
Standby Immediate				V			V	V	V		V
Write Buffer				V			V	V	V		V
Write Long	V		V	V			V	V	V		V
Write Multiple	V		V	V			V	V	V		V
Write Sector(s)	V		V	V			V	V	V		V
Invalid Command				V			V	V	V		V

NOTE:
V = Valid for this command

7

ABSOLUTE MAXIMUM RATINGS*

Operating Temperature[1] . . . . . 0°C to +60°C
Storage Temperature . . . . . −30°C to +70°C
Any Pin Voltage, from Ground[2] . . . −2.0V to +7.0V
V_{CC} Voltage, from Ground . . . . . −0.5V to +6.0V

NOTES:

1. Operating temperature is for commercial product defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to −2.0V for periods less than 20 ns. Maximum DC output-pin voltage is V_{CC} + 0.5V which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.

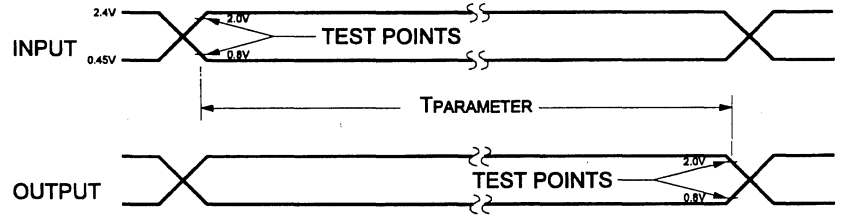
NOTICE: This data sheet contains information on products in the sampling and initial production phases of development. The specifications are subject to change without notice. Verify with your local Intel Sales office that you have the latest data sheet before finalizing a design.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

DC CHARACTERISTICS T_A = 0°C to +60°C, V_{CC} = 4.75V to 5.25V

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Condition
I_{LI}	Input Leakage Current			±1	±20	µA	V_{CC} = V_{CC} Max V_{IN} = V_{CC} or GND
I_{LO}	Output Leakage Current			±1	±20	µA	V_{CC} = V_{CC} Max V_{IN} = V_{CC} or GND
I_{CCP}	V_{CC} Peak Current at V_{PP} Startup			700		mA	
I_{CCR}	V_{CC} Read Current			130		mA	
I_{CCW}	V_{CC} Write Current			140		mA	
I_{CCE}	V_{CC} Reclamation (Erase) Current			140		mA	
I_{CCI}	V_{CC} Idle Current	1		2	80	mA	
I_{CCS}	V_{CC} Standby Current	2		2		mA	
I_{CCSL}	V_{CC} Sleep Current	1, 2, 3		2		mA	
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.0		V_{CC}+0.3	V	
V_{OL}	Output Low Voltage		V_{SS}		0.45	V	V_{CC} = V_{CC} Min
V_{OH}	Output High Voltage		2.4		V_{CC}	V	V_{CC} = V_{CC} Min

NOTES:
1. I_{CCI} Max current applies only when a host *idle* command forces the drive into idle mode. If no activity is detected after about 1.5 seconds, the drive automatically transitions to standby mode.
2. Standby and sleep modes are equivalent.
3. After command completion, the drive enters standby mode. It wakes and responds to a host command within 10 ms.



AC test inputs are driven at V_{OH} (2.4V) for a logic "1" and V_{OL} (0.45V) for a logic "0", Input timing begins at V_{IH} (2.0V) and V_{IL} (0.8V). Output timing ends at V_{IH} and V_{IL}. Input rise and fall times (10% to 90%) < 10 ns.

Figure 3. AC Input/Output Reference Waveform

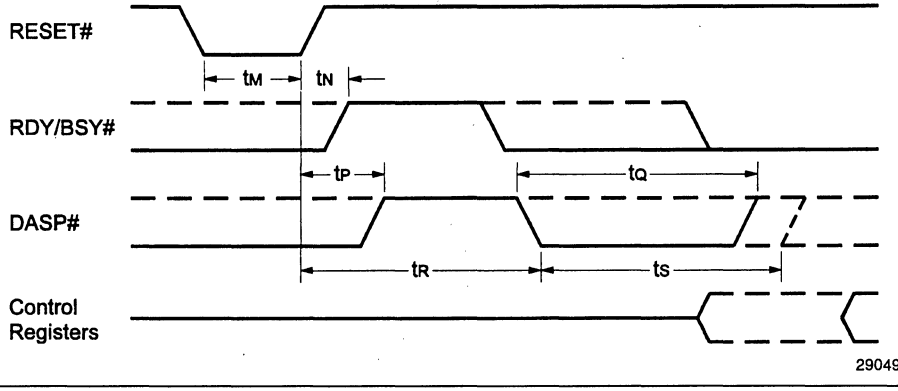


Figure 4. IDE-ATA Mode Reset Timing Diagram

Table 12. Power-On and Reset Timing Parameters

Symbol	IEEE	Parameter	Min	Max	Units
t_M	t_{SLSH}	RESET# Pulse Width	25		µs
t_N	t_{SHYH}	RESET# to BSY		400	ns
t_P	t_{SHDSPH}	RESET# to DASP# High		1	ms
t_Q	t_{DSHDSL}	DASP# Pulse Width		30	sec
t_R	t_{SHDSL}	RESET# to DASP# Low		450	ms
t_S	$t_{DSHDSLE}$	DASP# Pulse Width (Extended)		30.5	sec

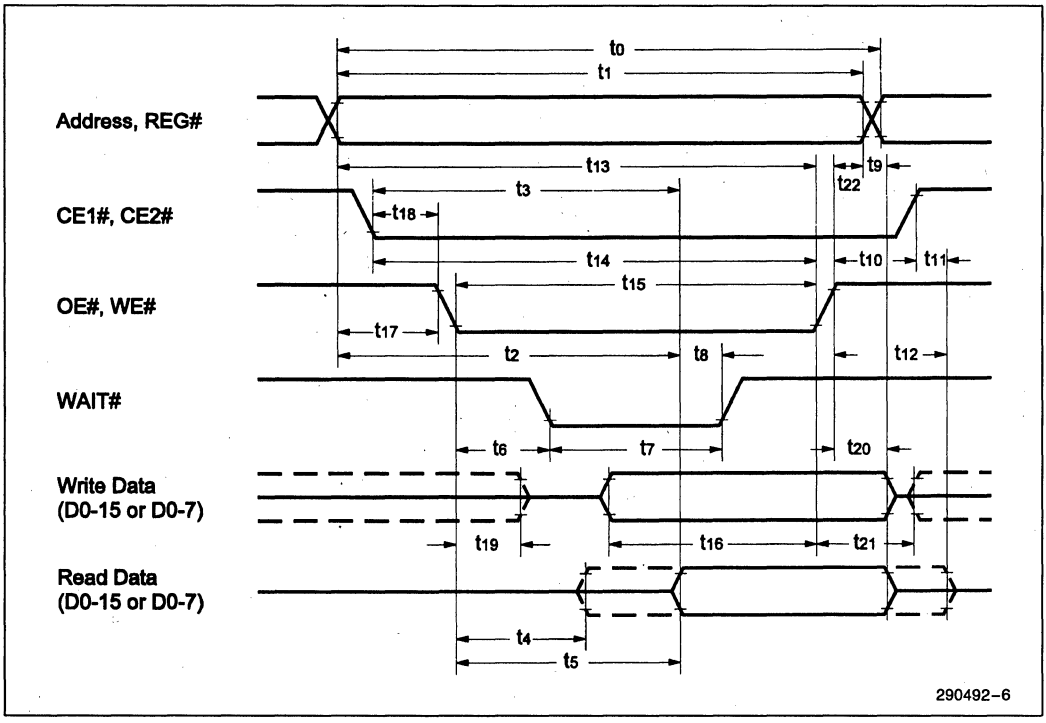


Figure 5. Attribute-Mode Timing Diagram

Table 13. Attribute-Mode Timing Parameters

Symbol	IEEE	Parameter	Min	Max	Units
READ					
t1	t_{AVAVR}	Read Cycle Time	300		ns
t2	t_{AVQV}	Address Access Time		300	ns
t3	t_{ELQV}	Card Enable Access Time		300	ns
t4	t_{GLQNZ}	Output Enable to Data Change	5		ns
t5	t_{GLQV}	Output Enable Access Time		150	ns
t6	t_{GLWtV}	Wait Valid from Output Enable		35	ns
t7	t_{WtLWtH}	Wait Pulse Width		12	µs
t8	t_{QVWtH}	Data Setup for Wait Released	0		ns
t9	t_{AXQX}	Data Valid after Address Change	0		ns
t10	t_{GHEH}	Card Enable Hold Time	20		ns
t11	t_{EHQZ}	Data Disable from Card Enable High		100	ns
t12	t_{GHQZ}	Data Disable from Output Enable High		100	ns

Table 13. Attribute-Mode Timing Parameters (Continued)

Symbol	IEEE	Parameter	Min	Max	Units
WRITE					
t0	t_{AVAVW}	Write Cycle Time	250		ns
t13	t_{AVWH}	Address Valid to Write Enable High	180		ns
t14	t_{ELWH}	Card Enable to Write Enable High	180		ns
t15	t_{WLWH}	Write Enable Pulse Width	150		ns
t16	t_{DVWH}	Data Valid to Write Enable High	80		ns
t17	t_{AVWL}	Address Valid to Write Enable Low	30		ns
t18	t_{ELWL}	Card Enable Setup to Write Enable Low	0		ns
t19	t_{WLQZ}	Write Enable to Previous Read-Data Disabled		100	ns
t20	t_{WHDX}	Data Hold after Write Enable High	30		ns
t21	t_{WHQNZ}	Data Driven after Write Enable High	5		ns
t22	t_{WHAX}	Address Hold after Write Enable High	30		ns

7

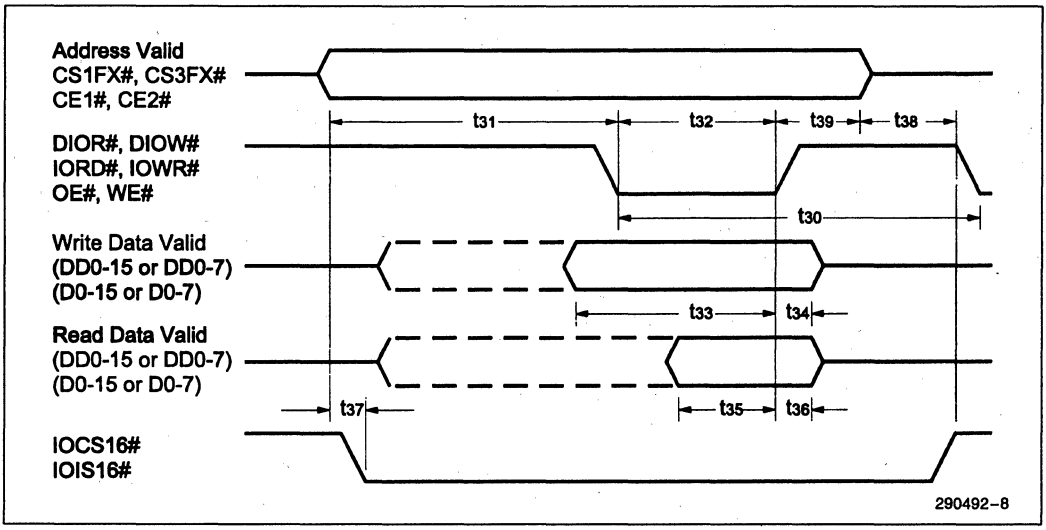


Figure 6. IDE-ATA and PCMCIA-ATA Programmed I/O and Memory Timing Diagram

Table 14. IDE-ATA and PCMCIA-ATA Programmed I/O and Memory Timing Parameters

Symbol	IEEE	Parameter	Min	Max	Units
t30	t_{IGLIGL}	Cycle Time	383		ns
t31	t_{AVIGL}	Address, Card Enable, to Read/Write Setup	50		ns
t32	t_{IGLIGH}	Read/Write Pulse Width	125		ns
t33	t_{DVIGH}	Data Setup to Write High	45		ns
t34	t_{IGHDX}	Data Hold after Write High	20		ns
t35	t_{QVIGH}	Data Setup to Read High	35		ns
t36	t_{IGHQX}	Data Hold after Read High	0	5	ns
t37	t_{AVISL}	Address or Card Enable Valid to IOCS16# Assertion	0	50	ns
t38	t_{AXISH}	Address or Card Enable Valid to IOCS16# Negation		45	ns
t39	t_{IGHAX}	Read/Write to Address or Card Enable Invalid	15		ns

INDEX

6
68-pin IDE-ATA, 5

A
abort, 27
AC input/output reference waveform, 37
address mark not found, 27
addressing configurations, 7
alternate status register, 28
altitude, 5
ATA (AT-Attachment)
  -IDE, 5
  68-pin, 5
  auto-detect mode, 4
  code update, 33
  command descriptions, 28
  compatibility, 26
  functions, 26
  interface for disk drives, 6
  INTRQ, 8
  mode, 7, 8, 11, 18
  OE# (IDE-DET#), 8, 9
  power management, 4
  RESET#, 8
  timing, 40
attribute mode
  accessing, 19
  plane, 11, 25
  timing, 38
auto
  power-down, 33
  sleep, 33

B
bad block mark, 26
block transfer, 31
BSY (busy), 28, 30, 31, 32
buffer, 26
  size, 5, 34
  type, 34
bytes per sector or track, 34

C
cache, 33
capacity, 5
card
  configuration and status register, 25
  information structure (CIS), 19
check power mode, 29, 31, 35
CHS (cylinder, head, sector), 26, 27, 34
clean-up
  state machine (CSM), 33
  time, 33
command
  aborted, 31
  codes, 29
  governing, 33
  invalid, 35
  register, 28
common memory plane, 11
compression, 33
configuration, 7
  index number, 25
  option register, 19, 25
  table entry tuple
    I/O space definition, 21
    memory space definition, 21
    primary drive address:, 21
    secondary drive address, 22
  tuple, 20
corrected data, 28, 31
corrupted data, 31
cylinder, 5
  current, 34
  high register, 27, 30
  low register, 27, 30
  number, 34
  registers, 29, 33

D
data bus, 18
data register, 26
DC characteristics, 36
device information tuple, 20
diagnostic
  code, 26, 30
  tests, 30
disk
  device interface function extension tuple, 23
  errors, 31
  transfer rate, 34
DMA, 34
doubleword I/O, 34
DRDY (drive ready), 28
drive
  /head register, 27, 29, 31, 33
  address register, 28
  control register, 28
  identify, 32, 34, 35
  parameters, 32
  select, 27, 28
DRQ (data request), 28, 30, 31, 32
DSC (drive seek complete), 28
DWF (drive write fault), 28

E
ECC (error correction code), 4, 5, 18, 19, 30, 33
environment, 5, 6

error
  command aborted, 32
  correction, 6
  ERR, 26, 28
  non-recoverable rate, 6
  posting, 33, 35
  rate, 6
  read rate, 5
  register, 26, 30, 35
even-register contents, 18
execute drive diagnostic, 26, 28, 29, 30, 35

F

failure rate, 6
faulty-part replacement, 6
features get/set, 31
features register, 29
firmware rev., 34
flash cycling, 6
format capacity, 5
format track, 26, 27, 28, 29, 30, 35
function ID tuple for disk function, 23

G

get features, 31, 33

H

hardware
  interface, 6
  reset, 6, 31
head select number, 28
heads, 5
  current, 34
  number, 34
high-byte data, 18
HRST# (host reset), 31
humidity, 5

I

I/O space definition, 21
ID-field not found, 26
IDE-ATA
  (AT-Attachment), 5
  68-pin, 5
  auto-detect mode, 4
  command descriptions, 28
  compatibility, 26
  functions, 26
  interface for disk drives, 6
  INTRQ, 8
  mode, 7, 8, 11, 18
  OE# (SELATA#), 8, 9
  power management, 4
  RESET#, 8
  timing, 40
identify drive, 26, 28, 29, 32, 34, 35
idle, 6, 29, 31, 35, 36
  immediate, 29, 31, 35
  mode, 31
independent I/O mode, 6, 14, 15, 18
independent memory mode, 13, 14, 18
index, 28
initialize drive parameters, 27, 29, 31, 35
interface, 5
  hardware, 6
  transfer rate, 5
interrupt, 28, 30, 31, 32
  acknowledge, 28
  enable (IEN#), 28
  level mode, 25
  pulse mode, 25
  signal, 25
IOIS16#, 18, 19
IREQ#/INTRQ, 8

J

JEDEC programming information tuple for common memory, 23

L

LBA (logical block address), 26, 27, 30, 34
level 1 version/product information tuple, 24
level mode interrupt, 25
low-byte data, 18
low-level format, 33

M

manufacturer identification tuple, 23
master/slave, 33
maximum ratings, 36
media
  change, 26
  transfer rate, 5
memory
  mode, 7
  space definition, 20
model no., 34
MTBF (mean time between failures), 6
multiple sector, 34

N

no-link control tuple (end of tuples marker), 24
noise, 5
non-operating temperature, 5
non-recoverable error rate, 6
number
  of cylinders, 34
  of heads, 34

O

odd-register contents, 18
OE# (SELATA#), 8, 9
operating temperature, 5, 36
option index number, 19
ordering information, 45

P
PC card ATA features function extension tuple, 23
PCMCIA
  card configuration registers, 25
  memory, I/O signal, 8
  modes, 18
  PC card ATA specification 1.02, 4, 5
  PC card standard release 2.1, 4, 5, 6
  power management, 4
  Type-2 case dimensions, 7
PCMCIA-ATA
  addressing modes, 25
  compliant, 4
  configurations, 19
  IREQ#, 8
  modes, 7, 8
  reset, 8
performance, 5
physical characteristics, 6
pin description, 9
  A0–24, 9
  BVD1, BVD2, 10
  CD1#, CD2#, 10
  CE1#, CE2#, 10
  CS1FX#, 9
  CS3FX#, 10
  D0–15, 9
  DA0–2, 9
  DASP#, 10
  DD0–15, 9
  DIOR, 10
  DIOW, 10
  DMACK#, 10
  DMARQ, 10
  GND, 9
  IDE_DET#, 9
  INPACK#, 10
  IOCS16#, 9
  IOIS16#, 9
  IORD#, 10
  IORDY, 10
  IOWR#, 10
  IREQ#/INTRQ, 9
  OE#, 9
  PDIAG, 10
  RDY/BSY#, 9
  REG#, 10
  RESET, 10
  RESET#, 10
  RFSH#, 10
  STSCHG#, 10
  V_{CC}, 9
  V_{PP1}, V_{PP2}, 9
  WAIT#, 10
  WE#, 9
pin replacement register, 25, 26
pinout, 8
PIO (programmed I/O), 18, 40
power, 5
  idle, 5
  management, 4
  mode, 33
  peak current, 5
  read, 5
  RMS, 5
  sleep mode, 5
  standby, 5
  supply voltage, 5
  write, 5
power-down, 4, 6
power-on, 26, 28, 31, 32, 37
  defaults, 33
  hours, 5
  reset, 25
  to ready, 5
preventive maintenance, 6
primary drive address, 7, 16, 17, 18, 21
pulse mode interrupt, 25

R
read, 27
  buffer, 28, 29, 32, 35
  DMA, 32
  error rate, 5
  long, 18, 19, 28, 29, 30, 35
  look-ahead, 33
  multiple, 29, 31, 34, 35
  sector, 29, 30, 31, 35
  verify, 28, 29, 30, 35
recalibrate, 28, 29, 30, 35
reclamation, 30
reliability, 5, 6
repair, 6
reset, 10, 26
  hardware, 6, 8, 10, 31
  mode, 33
  PCMCIA-ATA, 8
  power-on, 25
  soft, 25
  software, 6, 31
  SRST (software), 28
  timing, 37
rotational latency, 5
ruggedness, 5

7

S
secondary drive address, 7, 16, 17, 18, 22
sector, 5
  block sizes, 32
  buffer, 30
  capacity, 34
  count, 33
  count register, 27, 29, 30, 31, 32
  number, 33
  number register, 27, 29, 30
  per track, 34
security ID, 33
seek, 5, 28, 29, 30, 35
serial no., 34
set features, 27, 29, 31, 33, 34
set multiple mode, 29, 31, 32, 35
set sleep mode, 6, 29, 31
shock, 4, 5
size, 5
sleep mode, 5, 6, 35, 36
socket and copy register, 25, 26
soft reset, 25
software reset, 6, 31
special features, 31
spinup, 5, 31
SRST (software reset), 28
standby, 5, 6, 29, 31, 35
  immediate, 29, 31, 32
  mode, 6, 31, 36
starting sector number, 27
status code, 26
status register, 27, 35
storage temperature, 36
subfunction code, 31
supply voltage, 5

T
temperature, operating/non-operating, 5
track 0 not found, 27
transfer cycle timing mode, 34
transfer rate, 34
tuple, 20
  configuration table entry
  I/O space definition, 21
  memory space definition, 21
  primary drive address, 21
  secondary drive address, 22
  device information, 20
  disk device interface function extension, 23
  function ID for disk, 23
  JEDEC programming information for common memory, 23
  level 1 version/product information, 24
  manufacturer identification, 23
  no-link control (end of tuples marker), 24
  PC card ATA features function extension, 23

U
uncorrectable data error, 26
useful life, 6

V
vendor unique arguments, 33
vibration, 5
voltage
  any pin, 36
  maximum DC output, 36
  minimum DC input, 36
  overshoot, 36
  undershoot, 36
  V_{CC}, 36

W
weight, 5
write
  buffer, 28, 29, 32, 35
  cache, 33
  DMA, 32
  drive fault, 28
  gate, 28
  long, 18, 19, 28, 29, 30, 35
  multiple, 29, 32, 34, 35
  power, 5
  sector, 29, 30, 32, 35

Z
zero-based LBA, 26

ORDERING INFORMATION

iFD005P2SAXXXXX
iFD010P2SAXXXXX

Where:

i = Intel
FD = Flash drive
005, 010 = Density in Megabytes
P = PCMCIA-ATA
2 = PCMCIA Type-2 form factor
S = Power supply voltage = 5V
A = Revision or stepping number

REVISION HISTORY

Number	Description	
002	Page 8:	Card Pin 33 PCMCIA-ATA Mode changed from WP/IOIS16# to IOIS16#.
	Page 8 and 10:	Card Pin 59 IDE-ATA Mode changed from D.U. to IORDY.
	Page 8–10:	IDE-ATA mode pin signal names changed to ATA-specification names.
	Page 10:	CD1#, CD2#: pull-down resistors changed from 470Ω to ground.
	Page 11–17:	Mode tables changed to reflect 16-bit access.
	Page 20:	Card Information Structure (CIS) Tuples added.
	Page 27:	Write Precomp register name changed to Set Feature register.
	Page 28:	Alternate Status Register bit 3 changed from FRQ to DRQ.
	Page 29:	Intel Get/Set Features code 9A added.
	Page 31:	Standby and Idle command descriptions: Power-down sequence and count-down timer description removed.
	Page 33:	Set Features command codes added.
	Page 33:	Table 9, Intel Vendor Unique Arguments, added.
	Page 34:	Table 10, Identify Drive Argument, values added.
	Page 36:	DC Characteristics I_{CCR} changed to 130 mA.
	Page 37:	t_M (t_{SLSH}) parameter changed to 25 μs to conform to ATA specification.
		t_N (t_{SHYH}) parameter changed to 400 ns to conform to ATA specification.
	Page 39:	Table 13 parameters changed to conform to the PCMIA specification.
	Page 39:	I/O-Mode Timing Diagram removed.
	Page 40:	I/O-Mode Timing Parameters Table removed.
	Page 40:	Table 14 parameters changed to conform to ATA mode 1 specification.

7

Flash Drive Design Guide (for iFD005P2SA/ iFD010P2SA)

December 1994

Order Number: 297537-001

FLASH DRIVE DESIGN GUIDE
(for iFD005P2SA/iFD010P2SA)

CONTENTS

PAGE

INTRODUCTION 7-49
- Specification Summary 7-50
- Related Documents and Standards ... 7-51
- Sockets and Physical Dimensions 7-52
- Memory and I/O Modes Decoding 7-53
- I/O-Mode Addressing 7-58
- I/O-Mode Signals 7-58
- Switching Between Memory and I/O Modes 7-59
- Host Drivers 7-60

IDE-ATA INTERFACE 7-61
- Pin Assignments 7-62
- Pin Descriptions 7-63
- Interface Bus Timing 7-64
- I/O Transfer Description 7-65
- Master/Slave Compatibility 7-65
- Resets 7-65
- Interrupts 7-67
- 68-pin to 44/40-pin Conversion 7-68

PCMCIA INTERFACE 7-69
- Pin Assignments 7-69
- Pin Descriptions 7-70
- Interrupts 7-71
  - Pulse Mode 7-71
  - Level Mode 7-71
  - Sharing Interrupts 7-72
- Interface Bus Timing 7-72
  - PCMCIA Reset Timing 7-72
  - Attribute-Mode Timing 7-73
  - PCMCIA-ATA Memory and I/O Timing 7-75
- Configuration Options Indexes 7-75
- Card Configuration Registers 7-77
  - Configuration Option Register 7-77
  - Configuration & Status Register 7-77
  - Pin Replacement Register 7-78
  - Socket and Copy Register 7-78

CONTENTS

PAGE

- CIS (Tuples) 7-79
  - Device Information Tuple 7-79
  - Configuration Tuple 7-80
  - Card Configuration Table Entry Tuples 7-81
  - Function ID Tuple for Disk Function 7-85
  - Disk Interface Function Extension Tuple 7-86
  - ATA Features Function Extension Tuple 7-86
  - ATA JEDEC Identifier Tuple 7-87
  - Manufacturer Identification Tuple .. 7-87
  - Level 1 Version/Product Information Tuple 7-87
  - No-link Control Tuple 7-88
- Card Configuration 7-89
- Twin Card Support 7-89
- Intel 82365SL and Compatible Chips 7-89

ATA REGISTERS 7-90
- Data Register (Read/Write) 7-90
- Error Register (Read) 7-91
- Set Feature Register (Write) 7-91
- Sector Count Register (Read/Write) .. 7-91
- Sector Number Register (Read/Write) 7-92
- Cylinder Low Register (Read/Write) .. 7-92
- Cylinder High Register (Read/Write) 7-92
- Drive/Head Register (Read/Write) 7-92
- Status Register (Read) 7-93
- Command Register (Write) 7-93
- Alternate Status Register (Read) 7-93
- Drive Control Register (Write) 7-94
- Drive Address Register (Read) 7-94
- Addressing Mode Descriptions 7-94

CONTENTS

PAGE

ATA COMMANDS 7-95
- Recalibrate, 1X 7-95
- Read Sector(S), 20 OR 21 7-96
- Read Long, 22 or 23 7-96
- Write Sector(s), 30 or 31 7-96
- Write Long, 32 or 33 7-96
- Read Verify Sector(s), 40 or 41 7-96
- Format Track, 50 7-97
- Seek, 7X 7-97
- Execute Drive Diagnostic, 90 7-97
- Initialize Drive Parameters, 91 7-98
- Standby Immediate, 94 OR E0 7-98
- Idle Immediate, 95 OR E1 7-98
- Standby, 96 OR E2 7-98
- Idle, 97 OR E3 7-98
- Check Power Mode, 98 OR E5 7-99
- Set Sleep Mode, 99 OR E6 7-99
- Intel Get/Set Features, 9A 7-99
- Read Multiple, C4 7-100
- Write Multiple, C5 7-101
- Set Multiple Mode, C6 7-101
- Read DMA, C8 OR C9 7-101
- Write DMA, CA OR CB 7-101
- Read Buffer, E4 7-102
- Write Buffer, E8 7-102
- Identify Drive, EC 7-102
  - Identify Drive Arguments 7-103
- Set Features, EF 7-104
- Error Posting 7-104

PROTOCOL OVERVIEW 7-105
- PIO Data In Commands 7-105
  - PIO Read Command 7-105
  - PIO Read Aborted Command 7-105
- PIO Data Out Commands 7-106

CONTENTS

PAGE

  - PIO Write Command 7-106
  - PIO Write Aborted Command 7-106
- Non-Data Commands 7-107
- Miscellaneous Commands 7-107
- DOS Data Format 7-107
  - Disk Layout 7-108
  - Disk Partitioning 7-108
  - Boot Sector 7-109
  - FAT Tables 7-109
  - Root Directory 7-110
  - Disk Formatting 7-111
- LBA Mode 7-111
- CHS Mode 7-111

LOW POWER MODES 7-112
- Idle 7-112
- Standby 7-112
- Sleep 7-112
- PCMCIA Power-Down 7-112

SYSTEM CONSIDERATIONS 7-113
- Card Type 7-113
- Power Requirements 7-113
- Bootable PCMCIA Drive 7-113
- Card Insertion and Removing 7-113
- Dealing with Existing System Drives 7-114
- PCMCIA Drivers 7-114
  - Socket Services 7-115
  - Card Services 7-116
  - Resource Allocation Utilities 7-116
  - ATA Driver 7-116
- Intel PCMCIA- and IDE-ATA Drivers 7-116
  - ATADRV1 7-116
  - ATADRV2 7-117
  - ATADRV3 7-117

CONCLUSION 7-117

INTRODUCTION

PCMCIA-ATA cards are easily implemented in desktop, portable, and handheld systems. The PCMCIA specification has enabled inter-system plug-and-play. Adapter chips and system software are standard off-the-shelf items that make the system designers job very simple.

Most ISV (independent software vendor) PCMCIA suites have been written for Intel-architecture DOS- or Windows-based platforms. System designers who implement PCMCIA sockets in these systems can obtain ready-made solutions. PCMCIA solutions for embedded and non-Intel-architecture applications, however, may be harder to find.

This design guide, although it isn't a step-by-step cookbook, provides important information that will help you create a successful PCMCIA-ATA or IDE-ATA solution. It contains specific information about memory and I/O addressing options, IDE-ATA and PCMCIA-ATA interfaces, the ATA and PCMCIA registers, ATA commands, read and write protocols, DOS disk and file structures, low power modes, the PCMCIA-ATA software stack, Intel's PCMCIA- and IDE-ATA drivers, and other ATA and PCMCIA system considerations.

In addition to previously undocumented information, this design guide summarizes important features of the ATA-specification and PCMCIA-standard. It covers in more detail the specifications, registers, pin descriptions, card information structure (CIS), tuples, timing parameters, and other topics contained in Intel's flash drive data sheet.

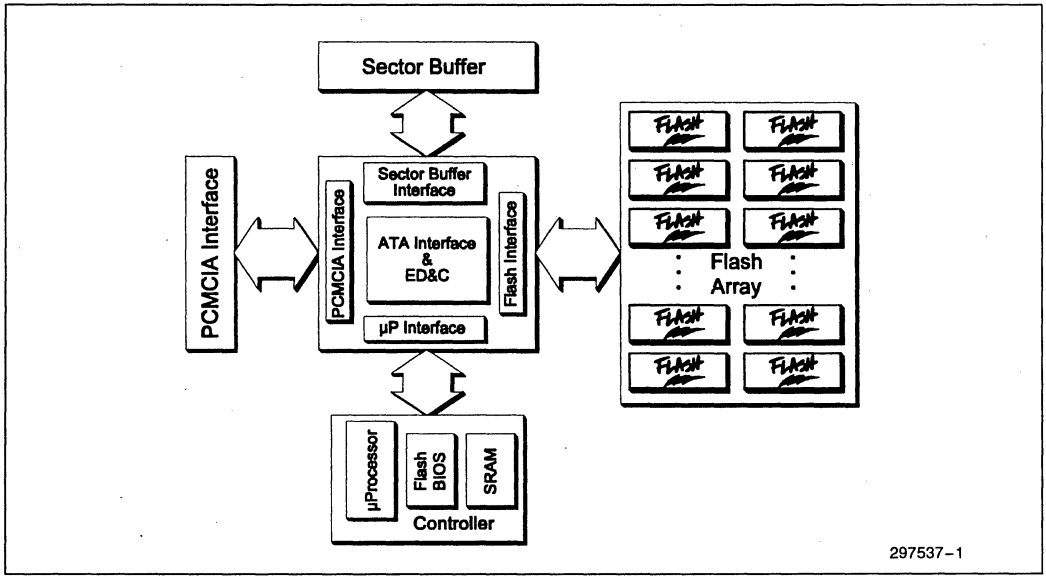


Figure 1. Flash Drive Block Diagram

7

Specification Summary
iFD005P2SA/iFD010P2SA Flash Drive

Capacity	(CHS Auto-Translate)
Formatted	Bytes (cylinders, heads, sectors)
iFD005, 5M	5,111,808 (156, 2, 32)
iFD010, 10M	10,485,760 (320, 2, 32)

Environmental

Temperature	
Operating	0°C to 60°C
Non-operating	−30°C to +70°C
Humidity, Operating	
(non-cond, 26°C wet bulb)	5% RH to 95% RH
Altitude, Operating	
(−200 ft to 40K ft)	−60m to 12 Km

Interfaces

68-pin PCMCIA-ATA
PCMCIA PC Card Standard, Release 2.1
PCMCIA PC Card ATA Specification 1.02
68-pin IDE-ATA 1.8" disk-drive-type ATA
ATA Interface for Disk Drives Standard

Performance

Seek Time (maximum or track-to-track)	< 1 ms
Rotational Latency	0 ms
Spinup Time	
Via Command (from deep power-down)	10 ms
Power-on to ready	2.5s (max)
Media Transfer Rate	
Read	8.0 MB/s
Write	0.27 MB/s
Interface Transfer Rate (burst)	5.0 MB/s
Buffer Size	32 KB

Power	(see DC specifications for details)
Read	0.5W
Write	0.7W
Idle	0.010W
Standby	0.010W
Sleep mode	0.010W
Peak current (programming voltage start-up)	0.7A
Supply Voltage	5.0V ± 5%

Reliability

MTBF	250K power-on-hours
(T_A = 25°C)	(25 KB write every 5 minutes)
	5M power-on-hours (typical)
Read Error Rate	
(with retries and ECC)	1 in 10^{14} bits read
ECC (optimized for flash memory)	32 bits/sector

Ruggedness	(any axis or direction)
Shock, op/non-operating	1000G
Vibration, op/non-operating	>15G, 10–500 Hz

Size	85.6mm x 54.0mm x 5.0mm (3.37" x 2.126" x 0.196")
Weight	29 gms (1.05 oz.)
Noise Absolutely silent operation	0 SPL (Sound Pressure Level)

Related Documents and Standards

This document contains information that supplements Intel's iFD005P2SA/iFD010P2SA Flash Drive data sheet, order number 290492.

Intel's Flash Drive is compatible with PCMCIA PC Card Standard, Release 2.1. To obtain this specification, write to:

PCMCIA
1030G East Duane Avenue
Sunnyvale, Ca 94086
Tel: (408) 720-0107
Fax: (408) 720-9416

The flash drive is also compatible with the "AT-Attachment Interface for Disk Drives" draft proposal. *See the internal working document X3T9.2, a Task Group of Accredited Standards Committee X3 of the American National Standard for Information Systems and the AT Attachment Specification for detailed information.* To obtain this specification, write to:

AT-Attachment Document Distribution
Global Engineering
15 Inverness Way East
Englewood, Co. 80112-5704
Tel: (800) 854-7179 or (303)792-2181
Fax: (303) 792-2192

The PC Card ATA Specification defines the electrical and register definitions that allow ATA-protocol mass storage devices to be used in PCMCIA-compliant sockets. It describes a superset specification that incorporates the ATA document (which defines the register set) and PC Card document (which describes the interface and electrical specifications). Socket hardware is PCMCIA compatible with certain pins redefined for I/O-mode compliance with PC Card document Release 2.01.

In default mode, the flash drive operates in PCMCIA-ATA compliant sockets. It conforms to PCMCIA PC Card standard 2.01, or higher (with ATA version 1.02 or higher hardware and software).

Its IDE-ATA auto-detect mode allows it to operate in standard IDE hardware- and BIOS-equipped systems (with 68-pin connector). It uses the industry-standard ATA command set so no additional software drivers are required.

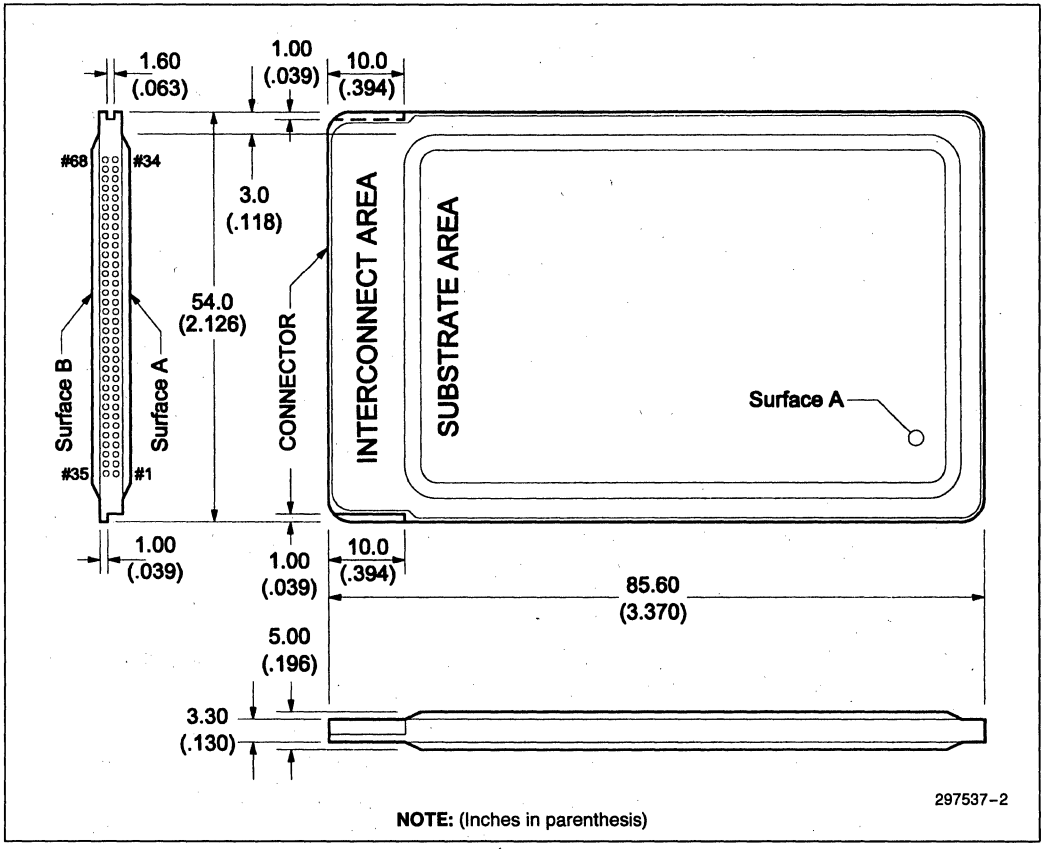
The flash drive supports the ATA-standard's primary and secondary drive address ranges in both PCMCIA- and IDE-ATA modes. This allows hosts to use standard disk-access BIOS routines.

Also supported is an independent I/O addressing mode that allows all ATA registers to be accessed within a contiguous 16- byte block.

Memory mapped I/O addressing is also supported. Hosts can access all ATA registers within the first 16 bytes of a 2-Kbyte contiguous memory block. In addition, the data register can be accessed by a host microprocessor's move-string instruction every time accesses occur between offsets 400h–7FFh. This mode provides PCMCIA PC Card Release 1.0 compatibility.

Sockets and Physical Dimensions

The flash drive's form factor is compatible with PCMCIA type-2 or type-3 compliant sockets. Figure 2 shows the drive's 68-pin PCMCIA Type-2 memory-card case dimensions.



NOTE: (Inches in parenthesis)

Figure 2. Flash Drive Dimensions

Memory and I/O Modes Decoding

Memory Mode Decoding

Memory mode allows hosts to access ATA registers within a 2-Kbyte contiguous memory space. The first 16 bytes contain ATA task-file registers. Offsets 400h–7FFh provide alternate ATA data register addresses. The host must decode memory addresses A_{11} and above to provide card-enables CE1# and CE2#. The drive decodes addresses A_{0-10}. The flash drive is in memory mode at power-on or card insertion or when the host writes configuration index value 00h to the card's Configuration Option register. Table 1 summarizes the memory-mode ATA task file registers.

Table 1. Memory-mode Task-file Registers

Register	Offset
Data	000h
Error (read), Set Feature (write)	001h
Sector Count	002h
Sector Number	003h
Cylinder Low	004h
Cylinder High	005h
Drive/Head	006h
Status (read), Command (write)	007h

Register	Offset
Data Low (duplicate)	008h
Data High (duplicate)	009h
Not Used	00Ah–00Ch
Error (duplicate)	00Dh
Alternate Status (read) Drive Control (write)	00Eh
Drive Address	00Fh
Data (duplicate)	400h–7FFh

8-bit-only hosts have trouble independently accessing the 16-bit data register's low and high bytes. Normally, an 8-bit host would access offset 000h for the low byte and 001h for the high byte. However, in ATA implementations, offset 001h is the error register. PCMCIA's PC Card ATA specification provides an alternate addressing scheme that solves this problem. The data register can be accessed at either offset 000h, 008h, 009h, or between offsets 400h–7FFh. The error register can be accesses at either offset 001h or 00Dh.

Offset 000h and 008h operate identically. In 8-bit mode (CE1# asserted, CE2# de-asserted, A_0 = 0), each 000h- or 008h- offset access presents sequential data bytes over D_{0-7}. The first access presents a data-word's low byte, the next access presents the word's high byte. Offset 000h accesses with CE1# de-asserted and CE2# asserted present Error register contents over D_{8-15}. In 16-bit mode (CE1# and CE2# asserted, A_0 = don't care), each offset 000h or 008h access presents sequential data words on D_{0-15}.

Offset 009h allows an 8-bit host to access the data register's high-byte over D_{0-7}. The drive presents low-then-high bytes if a host repeatedly accesses offset 008h followed by 009h. The drive presents high-then-low bytes if a host repeatedly accesses offset 009h followed by 008h. Sequential high-byte access at 009h are not supported.

Even-memory byte accesses (CE1# asserted, CE2# de-asserted, A_0 = 0) at offsets between 400h and 4FFh are equivalent to offset 008h accesses. Odd-memory byte accesses (CE1# asserted, CE2# de-asserted, A_0 = 1) at these offsets are equivalent to offset 009h accesses. Even-memory word accesses (CE1# and CE2# asserted, A_0 = don't care) at these offsets are equivalent to offset 000h or 008h word accesses.

Offsets 400h–7FFh do not correspond to actual card data-buffer addresses. Each access within this offset-range presents a new data byte or word from the data buffer's FIFO. This data-register aliasing allows host CPUs to perform memory string- move operations. In a read-from-drive example, the host can set a source pointer that corresponds to drive-offset 400h and a destination pointer that corresponds to internal RAM memory. A move-string instruction then transfers data from the source (flash drive) to RAM as it automatically increments source and destination pointers. Although the source pointer increments from 400h to 7FFh offsets, the drive internally accesses only the data register.

PCMCIA's PC Card standard describes other task file register accesses in the common memory plane. Truth Table 2 shows the PCMCIA-ATA independent memory address mode.

Table 2. PCMCIA-ATA MODE—Independent Memory address = 000–00F, 400–7FF(2)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–8	D7–0
No Operation	X(1)		X	H	H	X	X	X	X	High-Z	High-Z
Data (See Note 3)	000	Read	H	L	L	L	H	H	H	Data High	Data Low
			H	H	L	L	H	H	H	Invalid	Data
		Write	H	L	L	H	L	H	H	Data High	Data Low
			H	H	L	H	L	H	H	Invalid	Data
Error	001	Read	H	H	L	L	H	H	H	Invalid	Error
			H	L	H	L	H	H	H	Error	Invalid
Set Feature		Write	H	H	L	H	L	H	H	Invalid	Feature
			H	L	H	H	L	H	H	Feature	Invalid
Sector Count	002	Read	H	X	L	L	H	H	H	Invalid	Sect. Cnt.
		Write	H	X	L	H	L	H	H	Invalid	Sect. Cnt.
Sector Number	003	Read	H	H	L	L	H	H	H	Invalid	Sect. No.
			H	L	H	L	H	H	H	Sect. No.	Invalid
		Write	H	H	L	H	L	H	H	Invalid	Sect. No.
			H	L	H	H	L	H	H	Sect. No.	Invalid
Cylinder Low	004	Read	H	X	L	L	H	H	H	Invalid	Cyl. Low
		Write	H	X	L	H	L	H	H	Invalid	Cyl. Low
Cylinder High	005	Read	H	H	L	L	H	H	H	Invalid	Cyl. High
			H	L	H	L	H	H	H	Cyl. High	Invalid
		Write	H	H	L	H	L	H	H	Invalid	Cyl. High
			H	L	H	H	L	H	H	Cyl. High	Invalid
Drive/Head	006	Read	H	X	L	L	H	H	H	Invalid	Drv/Hd
		Write	H	X	L	H	L	H	H	Invalid	Drv/Hd
Status	007	Read	H	H	L	L	H	H	H	Invalid	Status
			H	L	H	L	H	H	H	Status	Invalid
Command		Write	H	H	L	H	L	H	H	Invalid	Command
			H	L	H	H	L	H	H	Command	Invalid
Data (Duplicate) (See Note 3)	008	Read	H	L	L	L	H	H	H	Data High	Data Low
			H	H	L	L	H	H	H	Invalid	Data
		Write	H	L	L	H	L	H	H	Data High	Data Low
			H	H	L	H	L	H	H	Invalid	Data
	009	Read	H	H	L	L	H	H	H	Invalid	Data High
			H	L	H	L	H	H	H	Data High	Invalid
		Write	H	H	L	H	L	H	H	Invalid	Data High
			H	L	H	H	L	H	H	Data High	Invalid
Invalid	00A-00C		H	X	X	X	X	X	X	Invalid	Invalid
Error (Duplicate)	00D	Read	H	H	L	L	H	H	H	Invalid	Error
			H	L	H	L	H	H	H	Error	Invalid
		Write	H	H	L	H	L	H	H	Invalid	Feature
			H	L	H	H	L	H	H	Feature	Invalid
Alternate Status	00E	Read	H	X	L	L	H	H	H	Invalid	Alt. Status
Drive Control		Write	H	X	L	H	L	H	H	Invalid	Control
Drive Address	00F	Read	H	H	L	L	H	H	H	Invalid	Drv. Add.
			H	L	H	L	H	H	H	Drv. Add.	Invalid
		Write	H	H	L	H	L	H	H	Invalid	Not Used
			H	L	H	H	L	H	H	Not Used	Invalid
Data (Duplicate) (See Note 3)	400-7FF	Read	H	L	L	L	H	H	H	Data High	Data Low
		Write	H	L	L	H	L	H	H	Data High	Data Low

NOTES:
1. X = don't care.
2. The host must decode addresses A_{11} and above to provide card enables CE1# and CE2# that place the drive on a 2-Kbyte boundary. The drive decodes A_{0-10}.
3. In independent memory mode, each byte access to 000, 008, or even addresses 400–7FE presents a new data-buffer byte on D_{0-7} starting with the first buffer word's low byte. Each word access presents sequential data buffer words on D_{0-15}. Byte access to 009 or odd addresses 401–7FF present high byte buffer data on D_{0-7} or D_{8-15} depending on CE1# and CE2#.

I/O Mode Decoding

Starting with PCMCIA's PC Card standard 2.0, system I/O space can be used to access PC cards. The flash drive provides three I/O addressing modes:

- independent I/O allows the drive to be mapped at any 16-byte I/O boundary. The host must decode I/O addresses A_4 and above to provide card-enables CE1# and CE2#. The drive decodes addresses A_{0-3}.
- primary drive address at system ATA I/O address 1F0h–1F7h and 3F6h–3F7. The host must decode I/O addresses A_{10} and above to provide card-enables CE1# and CE2#. The drive decodes addresses A_{0-9}.
- secondary drive address at system ATA I/O address 170h–177h and 376h–377. The host must decode I/O addresses A_{10} and above to provide card-enables CE1# and CE2#. The drive decodes addresses A_{0-9}.

Independent I/O mode is the most flexible because it allows hosts to place the drive at any available 16-byte I/O address boundary. A system can use multiple flash drives, limited only by available 16-contiguous-byte I/O spaces.

Primary and secondary drive addressing modes allow hosts to use the AT-standard's reserved disk drive I/O addresses. This provides PC system designers with the simplest way to accommodate ATA-protocol devices. Once the PCMCIA socket adapter is configured for the primary or secondary drive address ranges, a PC can take advantage of the BIOS's standard disk access routines.

Additionally, since the PC's host adapter must decode addresses A_{10} and above within its 64-Kbyte I/O space, up to 32 PCMCIA-ATA drivers can be placed at 4-kbyte boundaries. For example, drives can be placed at primary-type I/O addresses 01F0h, 11F0h, 21F0h, . . . F1F0h. Drives can also be placed at secondary-type I/O addresses 0170h, 1170h, 2170h, . . . F170h.

Truth Table 3 shows PCMCIA-ATA I/O address modes.

Table 3. PCMCIA-ATA MODE—Independent I/O address = xx0–xxF(2)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–8	D7–0
No Operation	X(1)		X	H	H	X	X	X	X	High-Z	High-Z
Data	0	Read	L	H	L	H	H	L	H	Invalid	Data Low
			L	L	L	H	H	L	H	Data High	Data Low
		Write	L	H	L	H	H	H	L	X	Data Low
			L	L	L	H	H	H	L	Data High	Data Low
Error	1	Read	L	H	L	H	H	L	H	Invalid	Error
			L	L	H	H	H	L	H	Error	Invalid
Set Feature		Write	L	H	L	H	H	H	L	X	Feature
			L	L	H	H	H	H	L	Feature	X

7

Table 3. PCMCIA-ATA MODE—Independent I/O address = xx0–xxF(2) (Continued)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–8	D7-0
Sector Count	2	Read	L	H	L	H	H	L	H	Invalid	Sect. Cnt.
			L	L	L	H	H	L	H	Sect. No.	Sect. Cnt.
		Write	L	H	L	H	H	H	L	X	Sect. Cnt.
			L	L	L	H	H	H	L	Sect. No.	Sect. Cnt.
Sector Number	3	Read	L	H	L	H	H	L	H	Invalid	Sect. No.
			L	L	H	H	H	L	H	Sect. No.	Invalid
		Write	L	H	L	H	H	H	L	X	Sect. No.
			L	L	H	H	H	H	L	Sect. No.	X
Cylinder Low	4	Read	L	H	L	H	H	L	H	Invalid	Cyl. Low
			L	L	L	H	H	L	H	Cyl. High	Cyl. Low
		Write	L	X	L	H	H	H	L	X	Cyl. Low
			L	X	L	H	H	H	L	Cyl. High	Cyl. Low
Cylinder High	5	Read	L	H	L	H	H	L	H	Invalid	Cyl. High
			L	L	H	H	H	L	H	Cyl. High	Invalid
		Write	L	H	L	H	H	H	L	X	Cyl. High
			L	L	H	H	H	H	L	Cyl. High	X
Drive/Head	6	Read	L	H	L	H	H	L	H	Invalid	Drv/Hd
			L	L	L	H	H	L	H	Status	Drv/Hd
		Write	L	H	L	H	H	H	L	X	Drv/Hd
			L	L	L	H	H	H	L	Command	Drv/Hd
Status	7	Read	L	H	L	H	H	L	H	Invalid	Status
			L	L	H	H	H	L	H	Status	Invalid
Command		Write	L	H	L	H	H	H	L	X	Command
			L	L	H	H	H	H	L	Command	X
Data (Duplicate)	8	Read	L	H	L	H	H	L	H	Invalid	Data Low
			L	L	L	H	H	L	H	Data High	Data Low
		Write	L	H	L	H	H	H	L	X	Data Low
			L	L	L	H	H	H	L	Data High	Data Low
	9	Read	L	H	L	H	H	L	H	Invalid	Data High
			L	L	H	H	H	L	H	Data High	Invalid
		Write	L	H	L	H	H	H	L	X	Data High
			L	L	H	H	H	H	L	Data High	X
Invalid	A-C		L	X	X	X	X	X	X	Invalid	Invalid
Error (Duplicate)	D	Read	L	H	L	H	H	L	H	Invalid	Error
			L	L	H	H	H	L	H	Error	Invalid
		Write	L	H	L	H	H	H	L	X	Feature
			L	L	H	H	H	H	L	Feature	X
Alternate Status	E	Read	L	H	L	H	H	L	H	Invalid	Alt. Status
			L	L	L	H	H	L	H	Drv. Add.	Alt. Status
Drive Control		Write	L	H	L	H	H	H	L	X	Control
			L	L	L	H	H	H	L	Not Used	Control
Drive Address	F	Read	L	H	L	H	H	L	H	Invalid	Drv. Add.
			L	L	H	H	H	L	H	Drv. Add.	Invalid
		Write	L	H	L	H	H	H	L	X	Not Used
			L	L	H	H	H	H	L	Not Used	X

PCMCIA-ATA MODE—Primary Address = 1F0–1F7, 3F6–3F7; Secondary Address = 170–177, 376–377(3)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–8	D7-0
No Operation	X		X	H	H	X	X	X	X	High-Z	High-Z
Data	1F0/170	Read	L	H	L	H	H	L	H	Invalid	Data Low
			L	L	L	H	H	L	H	Data High	Data Low
		Write	L	H	L	H	H	H	L	X	Data Low
			L	L	L	H	H	H	L	Data High	Data Low
Error	1F1/171	Read	L	H	L	H	H	L	H	Invalid	Error
			L	L	H	H	H	L	H	Error	Invalid
Set Feature		Write	L	H	L	H	H	H	L	X	Feature
			L	L	H	H	H	H	L	Feature	X
Sector Count	1F2/172	Read	L	H	L	H	H	L	H	Invalid	Sect. Cnt.
			L	L	L	H	H	L	H	Sect. No.	Sect. Cnt.
		Write	L	H	L	H	H	H	L	X	Sect. Cnt.
			L	L	L	H	H	H	L	Sect. No.	Sect. Cnt.
Sector Number	1F3/173	Read	L	H	L	H	H	L	H	Invalid	Sect. No.
			L	L	H	H	H	L	H	Sect. No.	Invalid
		Write	L	H	L	H	H	H	L	X	Sect. No.
			L	L	H	H	H	H	L	Sect. No.	X
Cylinder Low	1F4/174	Read	L	H	L	H	H	L	H	Invalid	Cyl. Low
			L	L	L	H	H	L	H	Cyl. High	Cyl. Low
		Write	L	H	L	H	H	H	L	X	Cyl. Low
			L	L	L	H	H	H	L	Cyl. High	Cyl. Low
Cylinder High	1F5/175	Read	L	H	L	H	H	L	H	Invalid	Cyl. High
			L	L	H	H	H	L	H	Cyl. High	Invalid
		Write	L	H	L	H	H	H	L	X	Cyl. High
			L	L	H	H	H	H	L	Cyl. High	X
Drive/Head	1F6/176	Read	L	H	L	H	H	L	H	Invalid	Drv/Hd
			L	L	L	H	H	L	H	Status	Drv/Hd
		Write	L	H	L	H	H	H	L	X	Drv/Hd
			L	L	L	H	H	H	L	Command	Drv/Hd
Status	1F7/177	Read	L	H	L	H	H	L	H	Invalid	Status
			L	L	H	H	H	L	H	Status	Invalid
Command		Write	L	H	L	H	H	H	L	X	Command
			L	L	H	H	H	H	L	Command	X
Invalid	3F/370-5	R/W	L	X	X	X	X	X	X	Invalid	Invalid
Alternate Status	3F6/376	Read	L	H	L	H	H	L	H	Invalid	Alt. Status
			L	L	L	H	H	L	H	Drv. Add.	Alt. Status
Drive Control		Write	L	H	L	H	H	H	L	X	Control
			L	L	L	H	H	H	L	Not Used	Control
Drive Address	3F7/377	Read	L	H	L	H	H	L	H	Invalid	Drv. Add.
			L	L	H	H	H	L	H	Drv. Add.	Invalid
		Write	L	H	L	H	H	H	L	X	Not Used
			L	L	H	H	H	H	L	Not Used	X

NOTES:
1. X = don't care.
2. The host must decode addresses A_4 and above to provide card enables CE1# and CE2# that place the drive on a 16-byte boundary. The drive decodes A_{0-3}.
3. The drive fully decodes these addresses using addresses A_{0-9}.

7

I/O-Mode Addressing

As mentioned above, the flash drive can be placed at standard PC ATA disk drive I/O addresses 1F0h and 170h. If a system's ATA adapter or fixed disk occupies these addresses, the PCMCIA adapter, or software driver that configures it, should avoid placing the flash drive at these addresses.

Standard AT BIOS-compatible systems place both the primary (or secondary) hard disk's Drive Address register and the primary (or secondary) floppy disk controller's Disk Change bit (bit 7) at I/O address 3F7h (or 177h). These devices normally don't conflict when 3F7h (or 377h) is accessed because the hard disk floats D_7 and the floppy floats D_{0-6}. However, PCMCIA hardware adapters, when configured for primary (or secondary) drive address, may be unable to float D_7 when 3F7h (or 377h) is accessed. One of the following methods can avoid this conflict:

- locate the flash drive at any I/O address mentioned earlier that doesn't conflict with 3F7h (or 377h).
- support only PCMCIA socket peripheral expansion. Don't support ATA or floppies via a dedicated controller card.
- don't allow access to address 3F7h (or 177h). Drivers that set up the adapter should use I/O windows that access 1F0h–1F7h and 3F6h (or 170h–177h and 376h). The driver should also disallow Drive Address register accesses.
- design system hardware that floats D_7 and de-asserts IOIS16# (the ISA and EISA bus' IOCS16#) when 3F7h (or 177h) is accessed by an ATA card. The system must float D_{0-6} and de-assert IOIS16# when 3F7h (or 177h) is accessed if a floppy drive is supported by the PCMCIA socket. This method requires special socket hardware but doesn't require modifications to existing BIOS software that accesses 3F7h (or 177h).

I/O-Mode Signals

With the introduction of PC Card standard Release 2.0, several memory-interface signals have alternate functions when used in I/O mode. These signals are:

- **RESET**—clears the Card Configuration Option register, which also initiates card initialization, and places the card in an unconfigured, memory-only interface state. The system must place RESET at high-Z for 1 ms (min) during card insertion or V_{CC} power-up. The flash drive internally pulls RESET to V_{CC} with a 100K resistor to ensure reset at card insertion and power-on. The card remains unconfigured until a valid configuration is written to the card's Configuration Option register.
- **WAIT#**—when asserted by the flash drive, extends an in-progress memory- or I/O-access cycle.
- **IORD**—when asserted with REG# and CE1# and/or CE2#, allows reads from card I/O space. The card must be configured (via the Card Configuration Option register) for one of its I/O modes.
- **IOWR#**—when asserted with REG# and CE1# and/or CE2#, allows writes to card I/O space. The card must be configured (via the card's Configuration Option register) for one of its I/O modes.
- **INPACK#**—asserted tells the host that the drive responds to I/O reads at the present address (CE1# and/or CE2# must be asserted). INPACK# enables host input data buffers. It is inactive until the host configures an I/O mode.
- **IOIS16#—(replaces memory mode WP)**—is asserted when the drive detects an address within its I/O address range that can present 16-bit data; card enables CE1# and CE2# are don't care. The host decides whether it accesses the data as 16- or 8-bits. If IOIS16# is not asserted, the host must access data as bytes.
  In IDE-ATA mode, this signal is IOCS16#. When asserted (low), the host must perform a 16-bit access. When not asserted, the host must perform an 8-bit access.

- **IREQ# (replaces memory mode RDY/BSY#)**—is asserted when the drive requests system service or at command/event completion. Primary and secondary address modes recommend, via the CIS tuples (explained later), that the host route IREQ# to IRQ_{14}, the standard PC-AT ATA interrupt. The adapter or software driver can ignore this recommendation and route the interrupt to any available CPU interrupt. Independent I/O mode allows the host to chose any available interrupt between IRQ_0 and IRQ_{15}.
  Level and pulsed interrupts are supported. The host can choose its preferred mode by writing to the Configuration Option register. Pulsed interrupts, usually used by ISA PC-architecture systems, are active-going-low with the host detecting the interrupt on the rising edge. Level interrupts, used in micro-channel PC-architecture and non-PC-compatible systems, are asserted low until the host services the interrupt. IREQ# is then de-asserted high.
  The card is in memory mode prior to configuration or after the host places the drive in memory mode via the Configuration Option register. This pin serves as a RDY/BSY# (ready/busy#) signal that informs the host when it can provide or accept information.
- **SPKR# (replaces memory mode BVD2)**—The flash drive does not support either the speaker (SPKR#) or battery voltage detect (BVD2) signals. In any PCMCIA mode, this signal is internally pulled to V_{CC} with a 10K resistor.
  In IDE-ATA mode, this signal is DASP# (host Drive-Active or Second-Present). DASP#-low at reset tells a master drive or host that a slave drive is present. During operation, DASP# can drive an LED to indicate that the drive is actively performing an operation.
- **STSCHG# (replaces memory mode BVD1)**—alerts the host, while the drive is in I/O mode, that memory-mode signals RDY/BSY# or WP changed state. Since the flash drive does not support these signals while I/O-mode configured, STSCHG# should be ignored.
  In IDE-ATA mode, this signal is PDIAG#. A slave drive uses PDIAG# to tell a master drive that it passed its diagnostic routines at power-on or reset.

Switching Between Memory and I/O Modes

Signal conflicts can be avoided if the following sequence is followed when switching the flash drive from its initial memory-mode (or unconfigured) state to one of the I/O operating modes.

- Disable adapter windows, I/O-related interrupts, and socket-generated interrupts caused by WP (IOIS16#), RDY/BSY# (IREQ#), BVD1 (PDIAG#), and BVD2 (DASP#).
- Assert REG# and read CIS (tuple) information to determine the drives configurable modes.
- Choose an option that fits the hosts memory, I/O, and interrupt capabilities. Initialize the card's Configuration and Status, Pin Replacement, and Socket and Copy registers. Write the appropriate option's index and desired interrupt mode (pulsed or level) to the Configuration Option register. If an I/O option is selected, the drive will be in I/O mode while the socket is configured for memory.
- Establish the socket's I/O related functions.
- Configure the socket for I/O operation.
- Route socket- and card-generated interrupts to desired CPU IRQs.
- Enable the socket's interrupts.

When a card is removed, use the following process to change the socket from an I/O to a memory configuration.

- Disable the socket's interrupts.
- If appropriate, undo the routing of socket- and card-generated interrupts from the CPU.
- Configure the socket for memory operation.
- Re-initialize the card's Configuration Option register with the memory-mode index (00h).
- Re-initialize and route socket-generated interrupts caused by WP (IOIS16#), RDY/BSY# (IREQ#), BVD1 (PDIAG#), and BVD2 (DASP#).
- Establish adapter windows.
- Enable adapter windows and memory-related interrupts.

Host Drivers

Intel's Flash Drive complies with both the PCMCIA-ATA and AT-Attachment specifications. Therefore, any standard AT BIOS and PCMCIA-ATA compliant drivers will work. Socket services, card services, and ATA drivers can be included in a system's BIOS or in a DOS machine's CONFIG.SYS.

Major industry BIOS ISVs provide PCMCIA-ATA support; four are listed below.

American Megatrends, Inc. (AMI)
6145-F Northbelt Parkway
Norcross, Ga. 30071
(404) 246-8625

Award software, Inc.
777 East Middlefield Road
Mountain View, Ca. 94043-4023
(415) 968-4433

Phoenix Technologies Ltd.
40 Airport Parkway
San Jose, Ca. 95110
(408) 452-6834

SystemSoft corp.
313 Speen St.
Natick, Ma. 01760
(508) 651-0088

Intel can provide various ATA drivers and utilities. Some work with existing socket and card services, some are stand-alone for DOS and non-DOS PCMCIA-equipped machines, others support various addressing modes (IDE-ATA, independent memory and I/O, and primary/secondary). See the Intel PCMCIA- and IDE-ATA Drivers section for more information.

IDE-ATA INTERFACE

In addition to PCMCIA-ATA modes, Intel's Flash Drive works in IDE-ATA-only implementations. PCMCIA-compliant sockets and drivers aren't required. The drive can work with a standard AT BIOS. Non-AT applications require only ATA-drive protocol BIOS drivers.

The flash drive determines that it's in an IDE-ATA socket, rather than a PCMCIA socket, by checking its OE# (SELATA#) pin. If SELATA# is grounded at reset (and thereafter), the drive is in IDE-ATA mode. Interface signals (Figure 3) and timing conform to the ATA specification. Truth Table 4 shows the IDE-ATA I/O address mode.

Table 4. IDE-ATA Mode—Primary = 1F0–1F7, 3F6–3F7; Secondary = 170–177, 376–377(2)

Function	ADDR	R/W	REG#	CE2#	CE1#	OE#	WE#	IORD#	IOWR#	D15–8	D7–0
No Operation	X(1)		X	H	H	X	X	X	X	High-Z	High-Z
Invalid	X		H	L	L	L	X	X	X	Invalid	Invalid
Data	0	Read	H	H	L	L	H	L	H	Data High	Data Low
		Write	H	H	L	L	H	H	L	Data High	Data Low
Error	1	Read	H	H	L	L	H	L	H	Invalid	Error
Set Feature		Write	H	H	L	L	H	H	L	Invalid	Feature
Sector Count	2	Read	H	H	L	L	H	L	H	Invalid	Sect. Cnt.
		Write	H	H	L	L	H	H	L	Invalid	Sect. Cnt.
Sector Number	3	Read	H	H	L	L	H	L	H	Invalid	Sect. No.
		Write	H	H	L	L	H	H	L	Invalid	Sect. No.
Cylinder Low	4	Read	H	H	L	L	H	L	H	Invalid	Cyl. Low
		Write	H	H	L	L	H	H	L	Invalid	Cyl. Low
Cylinder High	5	Read	H	H	L	L	H	L	H	Invalid	Cyl. High
		Write	H	H	L	L	H	H	L	Invalid	Cyl. High
Drive/Head	6	Read	H	H	L	L	H	L	H	Invalid	Drv/Hd
		Write	H	H	L	L	H	H	L	Invalid	Drv/Hd
Status	7	Read	H	H	L	L	H	L	H	Invalid	Status
Command		Write	H	H	L	L	H	H	L	Invalid	Command
Invalid	0–5		H	L	H	L	X	X	X	Invalid	Invalid
Alternate Status	6	Read	H	L	H	L	H	L	H	Invalid	Alt. Status
Drive Control		Write	H	L	H	L	H	H	L	Invalid	Control
Drive Address	7	Read	H	L	H	L	H	L	H	Invalid	Drv. Add.
		Write	H	L	H	L	H	H	L	Invalid	Not Used

NOTES:
1. X = don't care.
2. An AT-BIOS host adapter decodes these I/O addresses. The drive decodes A_{0-2}. OE#-low at reset selects IDE-ATA mode.

Pin Assignments

Table 5 shows IDE-ATA mode pin assignments.

Table 5. Flash Drive IDE-ATA Mode Pinout

Pin	IDE-ATA Mode	Notes	Pin	IDE-ATA Mode	Notes	Pin	IDE-ATA Mode	Notes	Pin	IDE-ATA Mode	Notes
1	GND		18	N.C.	1	35	GND		52	N.C.	1
2	DD_3		19	N.C.	1	36	N.U.	2	53	N.C.	1
3	DD_4		20	N.C.	1	37	DD_{11}		54	N.C.	1
4	DD_5		21	N.C.	1	38	DD_{12}		55	N.C.	1
5	DD_6		22	N.U.	2	39	DD_{13}		56	CSEL	3
6	DD_7		23	N.U.	2	40	DD_{14}		57	N.C.	1
7	CS1FX#		24	N.U.	2	41	DD_{15}		58	RESET#	4
8	N.U.	2	25	N.U.	2	42	CS3FX#		59	IORDY	
9	SELATA#	5	26	N.U.	2	43	N.C.	1	60	DMARQ	
10	N.C.	1	27	DA_2		44	DIOR#		61	DMACK#	
11	N.U.	2	28	DA_1		45	DIOW#		62	DASP#	
12	N.U.	2	29	DA_0		46	N.C.	1	63	PDIAG#	
13	N.C.	1	30	DD_0		47	N.C.	1	64	DD_8	
14	N.C.	1	31	DD_1		48	N.C.	1	65	DD_9	
15	N.C.	1	32	DD_2		49	N.C.	1	66	DD_{10}	
16	INTRQ	4	33	IOCS16#		50	N.C.	1	67	N.U.	2
17	V_{CC}		34	GND		51	V_{CC}		68	GND	

NOTES:
1. N.C. = Not connected internally.
2. N.U. = Connected internally but not used in mode.
3. CSEL replaces A25 in IDE-ATA mode to select master/slave configuration.
4. PCMCIA-ATA RESET and IREQ# polarities are RESET# and INTRQ in IDE-ATA mode.
5. SELATA# (OE#) determines PCMCIA-ATA (if high) or IDE-ATA (if low) mode during reset.

Pin Descriptions

Table 6. IDE-ATA Mode Pin Descriptions

Symbol	Pin	Direction	Description
GND	1,34,35,68		Ground between the host and drive.
DD_{0-15}	2-6,30-32, 37-41,64-66	I/O	16-bit bi-directional data bus. Registers and ECC accesses use DD_{0-7}. 16-bit data transfers use DD_{0-15}.
CS1FX#	7	I	Host card select 0 accesses command block registers.
DA_{0-2}	30–32	I	Drive addresses DA_{0-2} access registers or data port.
SELATA#	9	I	Detects IDE-ATA mode when externally grounded during and after reset.
INTRQ	16	O	Host enables Interrupt Request only when drive is selected and the host clears the Drive Control register's IEN#. INTRQ (a three-state pin) is high-Z when IEN# is set or the drive is not selected. INTRQ is set when the drive's CPU sets IRQ. INTRQ resets during Status register read or Command register write.
V_{CC}	17, 51	I	+5V DC supply to the drive.
IOCS16#	33	O	I/O-16 tells the host that the data register was accessed and the drive can send/receive 16-bit data. IOCS16# is a three-state pin.
CS3FX#	42	I	Host Chip Select 1 selects drive control block registers.
RFSH#	43	N.C.	Not used.
DIOR#	44	I	I/O Read enables drive register data onto DD_{0-15}. The host latches drive data on DIOR#'s rising edge.
DIOW#	45	I	I/O Write's rising edge strobes data into a drive register.
CSEL	56	I	CSEL determines whether the drive responds as master, when V_{IH}, or slave (not supported), when V_{IL}.
Reserved	57	N.C.	Reserved.
RESET#	58	I	At power-on or hardware reset, this active-low signal resets all drive registers.
IORDY	59	O	IORDY is not implemented.
DMARQ	60	O	Host DMA Request is not supported.
DMACK#	61	I	Host DMA ACKnowledge is not supported.
DASP#	62	O	Indicates Drive-Active or Second-Present when the drive is accessed. This open-drain output is pulled up with a 10 KΩ resistor.
PDIAG#	63	I/O	In slave mode (not supported), Passed Diagnostics (low) tells a master that diagnostics passed. In master mode, it's an input.

NOTE:
All pin directions are referenced to the drive.

Interface Bus Timing

In IDE-ATA mode, the flash drive conforms to the ATA specification's mode 1 programmed I/O bus timings. The drive address consists of DA_{0-2}, CS1FX#, and CS3FX#. Data consists of DD_{0-15} or DD_{0-7}. Figure 3 shows programmed I/O timing relationships and Table 7 shows the timing parameters.

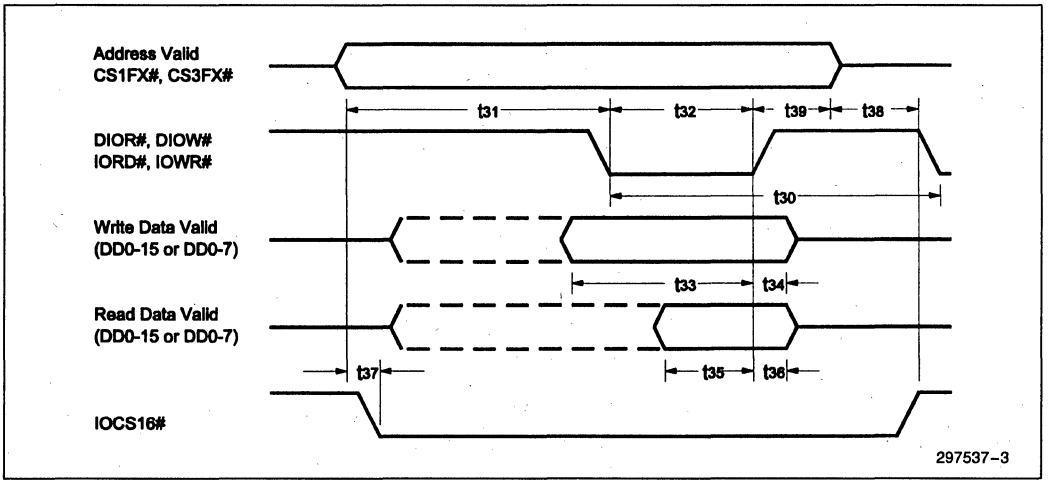


Figure 3. IDE-ATA Programmed I/O Timing Diagram

Table 7. IDE-ATA Programmed I/O Timing Parameters

Symbol	IEEE	Parameter	Min	Max	Units
t30	tIGLIGL	Cycle Time	383		ns
t31	tAVIGL	Address, CS1FX#, CS3FX#, to DIOR#/DIOW# Setup	50		ns
t32	tIGLIGH	DIOR#/DIOW# Pulse Width	125		ns
t33	tDVIGH	Data Setup to DIOW# High	45		ns
t34	tIGHDX	DIOW# Data Hold	20		ns
t35	tQVIGH	Data Setup to DIOR# High		35	ns
t36	tIGHQX	DIOR# Data Hold	0	5	ns
t37	tAVISL	Address, CS1FX#, CS3FX#, Valid to IOCS16# Assertion	0	50	ns
t38	tAXISH	Address, CS1FX#, CS3FX#, Valid to IOCS16# Negation		45	ns
t39	tIGHAX	DIOR#/DIOW# to Address, CS1FX#, CS3FX#, Invalid	15		ns

I/O Transfer Description

The host programs IDE-ATA drives to perform commands and return status at command completion. Input signals CS1FX#, CS3FX#, DA_{0-2}, DIOR#, and DIOW# select I/O registers that allow host-drive communication. The host uses Command Block registers to send commands or receive drive status and Control Block registers for drive control and to read the alternate status register. Table 8 lists the drive's IDE-ATA mode registers and their addresses.

Table 8. IDE-ATA-mode Task-file Registers

Register	CS1FX#	CS3FX#	DA_{0-2}
Data	L	H	0
Error (read) Set Feature (write)	L	H	1
Sector Count	L	H	2
Sector Number or LBA_{0-7}	L	H	3
Cylinder Low or LBA_{8-15}	L	H	4
Cylinder High or LBA_{16-23}	L	H	5

Register	CS1FX#	CS3FX#	DA_{0-2}
Drive/Head or LBA_{24-27}	L	H	6
Status (read) Command (write)	L	H	7
Alternate Status (read) Drive Control (write)	H	L	6
Drive Address	H	L	7
Invalid	L	L	X
Not Used	H	H	X
	H	L	0–5

Master/Slave Compatibility

Note that iFD005P2SA/iFD010P2SA drives do not support master/slave mode but future flash drives may.

The Drive/Head register's DRV bit selects the target drive; drive 0 = 0, drive 1 = 1. When daisy chained, the main drive is drive 0 and the other is drive 1. A single drive is drive 0.

When two drives are daisy chained, both receive commands, but only the selected drive executes them. However, both drives respond to an *execute diagnostics* command. Drive 1 posts its status to drive 0 via PDIAG#.

Throughout this document, drive selection refers to the DRV bit's state or the CSEL pin's state.

7

Resets

An IDE-ATA mode flash drive can be reset in three ways:

- At power-on, internal circuitry automatically resets the drive and places it in operating condition. The drive checks its electrical circuitry, initializes look-up tables, and sets default values.
- The host can assert (V_{IL}) RESET#. De-asserting RESET# (V_{IH}) places the drive in operating condition. The drive checks its electrical circuitry and resets default values.
- The host can set the Drive Control register's SRST (soft reset) bit. Clearing SRST places the drive in operating condition. The drive resets interface circuitry according to the Set Features requirement.

Figure 4 shows the IDE-ATA mode reset sequence and Table 9 shows reset timing parameters.

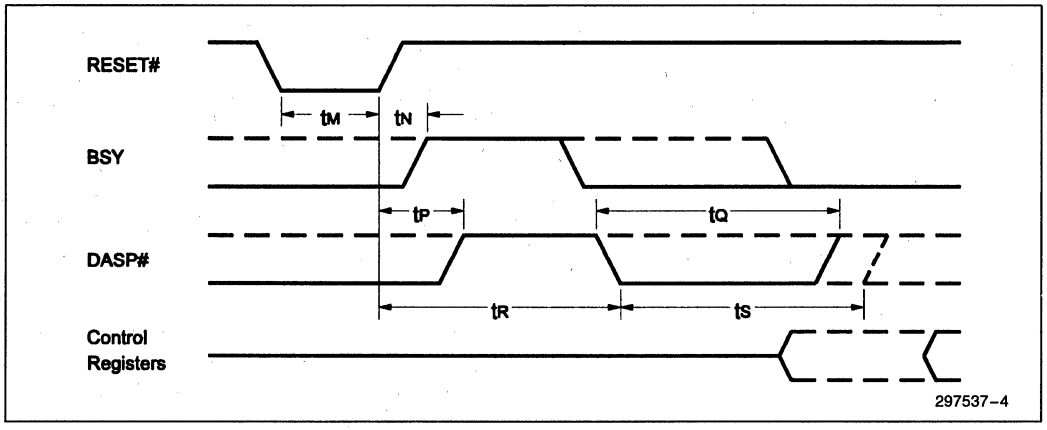


Figure 4. IDE-ATA Reset Timing Diagram

Table 9. IDE-ATA Power-On and Reset Timing Parameters

Symbol	IEEE	Parameter	Min	Max	Units
tM	tSLSH	RESET# pulse width	25		μs
tN	tSHYH	RESET# to BSY		400	ns
tP	tSHDSPH	RESET# to DASP# high		1	ms
tQ	tDSHDSL	DASP# pulse width		30	sec
tR	tSHDSL	RESET# to DASP# low		450	ms
tS	tDSHDSLE	DASP# pulse width (extended)		30.5	sec

The drive exits reset within 400 ns after RESET# negation or after setting the Device Control register's SRST. When the host asserts RESET#, drive 1 asserts DASP# within 400 ms to indicate its presence. DASP# remains asserted for 30 seconds or until drive 1 accepts the first command. When the host asserts SRST, the drive sets BSY = 1. When reset is accepted and BSY is set, both drives:

- perform hardware initialization
- clear previously programmed drive parameters
- may revert to default condition
- load default values into command block registers
- if a hardware reset, drive 0 waits for drive 1 to assert DASP#
- drive 1 asserts DASP# if operational
- if drive 1 asserts DASP#, drive 0 waits for PDIAG# assertion
- drive 1 clears BSY and PDIAG# if operational
- drive 0 clears BSY.

Interrupt is not generated at initialization-complete. If no self-tests are performed or no errors occur, default command block register values are:

- Error = 01h
- Sector Count = 01h
- Sector Number = 01h
- Cylinder Low = 00h
- Cylinder High = 00h
- Drive/Head = 00h

If a self-test is performed, the Error register contains a diagnostic code (see the *execute drive diagnostic* command). Following reset, the host should issue an *initialize drive parameters command.*

Interrupts

INTRQ interrupts the host only when the drive is selected, it has a pending interrupt, and the host has cleared the Device Control register's nIEN bit. If nIEN = 1, or the drive is not selected, INTRQ is high-Z, regardless of a pending interrupt's presence.

Any of the following will de-assert INTRQ:

- RESET# assertion
- setting the Device Control register's SRST
- host Command register writes
- host Status register reads

During PIO transfers, INTRQ is asserted at the beginning of each data block transfer, typically a single sector unless the *set multiple* command is used. *Format track, write sector(s), write buffer,* and *write long* commands are exceptions; INTRQ is not asserted at the beginning of the first data block transferred.

ISA XT and AT host interrupt controllers usually recognize positive-edge interrupts. The flash drive's IDE-ATA-mode generates a positive edge INTRQ which remains high until de-asserted as described above.

Pull-up resistors hold an ISA host's IRQx signals high when not driven. As long as the IRQx does not make a low-to-high transition, a constant high level will not generate an interrupt. When multiple IDE drives share one of these interrupts, only one drive at a time is allowed to assert its INTRQ. All other drives must high-Z their INTRQs. Then, only the active drive can make the low-to-high INTRQ transition.

68-pin to 44/40-pin Conversion

When operating in IDE-ATA mode, the flash drive's address, data, and control signals are routed directly from the 68-pin interface to equivalent signals on the 40-pin (typical of 3.5" HDDs) or 44-pin (typical of 2.5" HDDs) interface. The 68-pin socket's SELATA# (OE#) signal is connected to ground. This tells the flash drive that it's in an IDE-ATA socket rather than a PCMCIA-compliant socket. It also makes the drive's interrupt request (INTRQ) and reset (RESET#) signal polarities IDE-ATA compliant. Figure 5 shows a 68-to-40/44-pin adapter schematic. Contact Intel for source files that can be used or modified to build similar adapters.

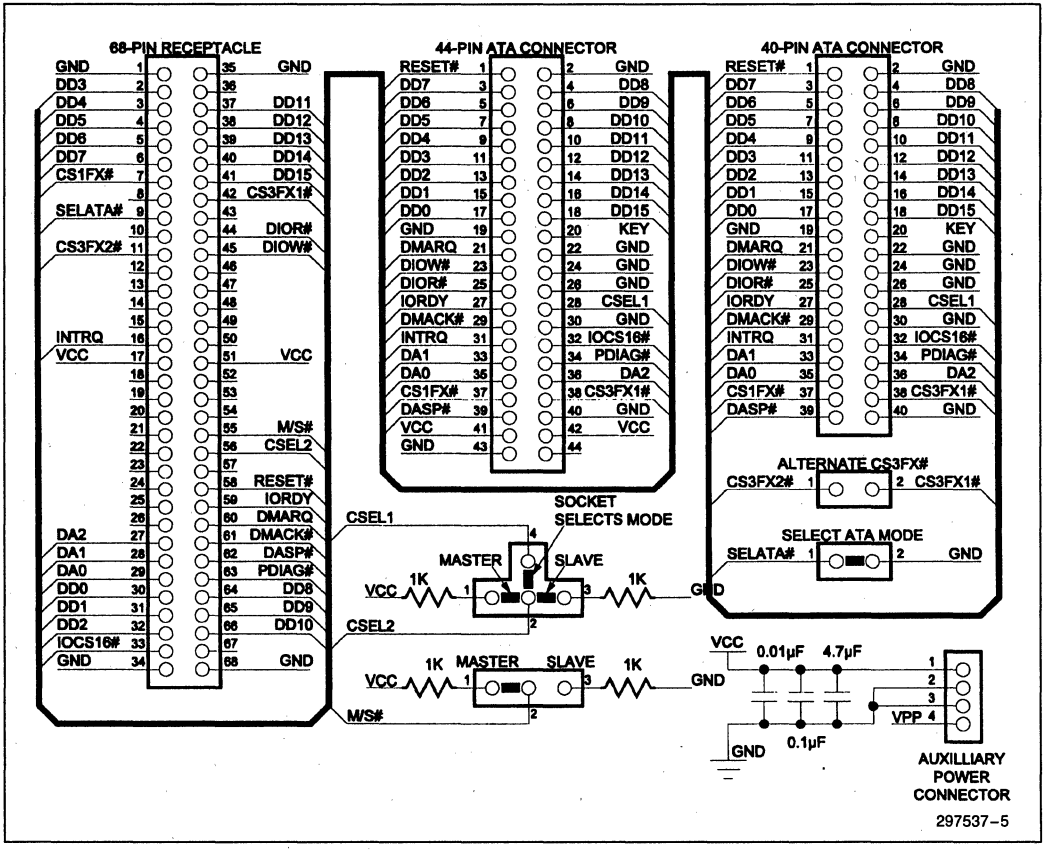


Figure 5. 68-to-40/44-Pin Adapter Schematic

PCMCIA INTERFACE

The PCMCIA standard, originally created to standardize memory-card to personal-computer interfaces, now includes I/O devices. It defines signals, voltage levels, and socket- and card-services that control hardware. The PC Card-ATA Mass Storage Standard combines PC card interface requirements with the industry-standard ATA specification.

In addition to primary and secondary ATA addressing modes, PCMCIA specifies linear, independent I/O and linear memory-mapped modes. The PCMCIA-ATA specification goes beyond the ATA specification's 16-bit-only data register and 8-bit-only task file register addressing protocol. It allows all ATA register accesses to be either 8-or 16-bits which allows a wide variety of system architectures to access PCMCIA-ATA drives.

Pin Assignments

Tables 10 and 11 show PCMCIA-ATA memory and I/O mode pin descriptions.

Table 10. Flash Drive PCMCIA Pinout

Pin	PCMCIA-ATA Memory Signal	PCMCIA-ATA I/O Signal	Notes
1	GND	GND	
2	D_3	D_3	
3	D_4	D_4	
4	D_5	D_5	
5	D_6	D_6	
6	D_7	D_7	
7	CE1#	CE1#	
8	A_{10}	N.U.	2
9	OE#	N.U.	2
10	N.C.	N.C.	1
11	A_9	A_9	2
12	A_8	A_8	2
13	N.C.	N.C.	1
14	N.C.	N.C.	1
15	WE#	N.U.	2
16	RDY/BSY#	IREQ#	
17	V_{CC}	V_{CC}	
18	N.C.	N.C.	1
19	N.C.	N.C.	1
20	N.C.	N.C.	1
21	N.C.	N.C.	1
22	A_7	A_7	
23	A_6	A_6	
24	A_5	A_5	
25	A_4	A_4	
26	A_3	A_3	
27	A_2	A_2	
28	A_1	A_1	
29	A_0	A_0	
30	D_0	D_0	
31	D_1	D_1	
32	D_2	D_2	
33	WP (N.U.)	IOIS16#	2
34	GND	GND	

Pin	PCMCIA Memory Signal	PCMCIA-ATA I/O Signal	Notes
35	GND	GND	
36	CD1#	CD1#	
37	D_{11}	D_{11}	
38	D_{12}	D_{12}	
39	D_{13}	D_{13}	
40	D_{14}	D_{14}	
41	D_{15}	D_{15}	
42	CE2#	CE2#	
43	N.C.	N.C.	1
44	N.U.	IORD#	2
45	N.U.	IOWR#	2
46	N.C.	N.C.	1
47	N.C.	N.C.	1
48	N.C.	N.C.	1
49	N.C.	N.C.	1
50	N.C.	N.C.	1
51	V_{CC}	V_{CC}	
52	N.C.	N.C.	1
53	N.C.	N.C.	1
54	N.C.	N.C.	1
55	N.C.	N.C.	1
56	N.U.	N.U.	2
57	N.C.	N.C.	1
58	RESET	RESET	
59	WAIT#	WAIT#	
60	INPACK#	INPACK#	
61	REG#	REG#	
62	Pullup	Pullup	
63	Pullup	STSCHG#	
64	D_8	D_8	
65	D_9	D_9	
66	D_{10}	D_{10}	
67	CD2#	CD2#	
68	GND	GND	

NOTES:
1. N.C. = Not connected internally.
2. N.U. = Connected internally but not used in mode.

Pin Descriptions

Table 11. PCMCIA-Mode Pin Description

Symbol	Pin	Direction	Description
GND	1, 34, 35, 68		Ground between the host and drive.
D_{0-15}	2-6, 30–32, 37–41, 64-66	I/O	Bi-directional data bus between the host and drive.
CE1#	7	I	Card Enable 1 enables even data bytes on D_{0-7}.
A_{0-25}	8, 10–14, 19–29, 46–50, 53–56	I	Addresses A_{0-10} access data and registers depending on the memory or I/O mode chosen by the host. Addresses A_{11-24} are not used.
OE#	9	I	Output Enable reads attribute- and memory-mode data on D_{0-15}.
WE#	15	I	Write Enable writes attribute- and memory-mode data on D_{0-15}.
RDY/BSY#/ IREQ#	16	O	Indicates internally timed activities status. Drive can accept host accesses when high. In memory mode, RDY/BSY# does not reflect the ATA status register's BSY bit.
V_{CC}	17, 51	I	+5V DC supply to the drive.
V_{PP1}, V_{PP2}	18, 52	N.C.	+12V DC programming supply is not required.
IOIS16#/WP	33	O	Held low after reset initialization (Write Protect is not supported).
CD1#, CD2#	36, 67	O	Card Detect pins, internally pulled low, allow the host to determine that the drive is fully inserted in the socket.
CE2#	42	I	Card Enable 2 enables odd data bytes on D_{8-15}.
RFSH#	43	N.C.	Not used.
IORD#	44	I	I/O Read gates data onto D_{0-15} during I/O-mode accesses.
IOWR#	45	I	I/O Write gates data from D_{0-15} during I/O-mode accesses.
Reserved	57	N.C.	Reserved.
RESET	58	I	Active-high power-on or hardware reset signal resets drive registers.
WAIT#	59	O	Signals that an in-progress host memory or I/O cycle is not complete.
INPACK#	60	O	When selected, the drive asserts Input Acknowledge while responding to an I/O read cycle.
REG#	61	I	Common memory is accessed when high. Attribute memory and ATA registers are accessed when low.
BVD2	62	O	10K Pull-up to V_{CC}.
BVD1/ STSCHG#	63	I/O	Notifies the host of RDY/BSY# and Write Protect state changes.

NOTE:
All pin directions are referenced to the drive.

Interrupts

IREQ# is an I/O-mode interrupt request signal; it is Ready/Busy # in memory mode. While in I/O mode, IREQ#, or polling the ATA Status register's Busy bit, signals command completion or data buffer availability. Although the card Pin Replacement Register's Rrdy/bsy # bit mirrors this signal, it is not typically used to determine ATA status.

Immediately following a host hardware reset or at power-on, the card is placed in a PCMCIA memory-mode interface. Once configured for an I/O interface mode, by writing a valid configuration option number to the card's PCMCIA Card Configuration register, the card's IREQ# signal can serve as a host interrupt.

IREQ# can interrupt the host. It's asserted only when the drive has a pending interrupt, the drive is selected, and the host previously cleared the Drive Control register's nIEN bit. If the drive is not selected or nIEN = 1, this output is high-Z, regardless of a pending interrupt's presence or absence.

IREQ# shall be negated by: reset assertion; setting the ATA Drive Control register's SRST bit; a Command register write; or Status register read. During PIO transfers, the drive asserts IREQ# at the beginning of each data block transfer. Except for pre-declared (by a *set multiple* command) *read-* or *write-multiple* operations, a single sector typically comprises a data block. However, IREQ# is not asserted at the beginning of the first data block transferred during *format track, write sector(s), write buffer, and write long* commands.

A host ATA Status register read clears IREQ#. A host ATA Alternate Status register read returns Status register information but does not clear IREQ#.

Intel's flash drive supports both Level- and Pulse-Mode interrupts. The PCMCIA interface specification defines Level- and Pulse-Mode interrupts that can be generated by a card's interrupt request signal. A host's hardware socket routes a PCMCIA card's interrupt request signal to the appropriate system interrupt.

Pulse Mode

Pulse-Mode interrupts, found in AT-attachment ISA bus implementations, handle edge-sensitive, "wire-AND" request lines. When it requests an interrupt, the flash drive forces a low-then-high IREQ# signal transition. The pulse's trailing edge generates the interrupt, per the PC Card specification.

8259-type interrupt controllers operating in the edge sensitive mode require three conditions on IRQn inputs: a positive-edge transition; a high state when the host's CPU acknowledges an interrupt; a pull-up resistor that keeps IRQn high when not driven by an interrupt source. Therefore, an AT bus system socket should not invert the PCMCIA Pulse-Mode IREQ# signal to produce the internal system bus' IRQn signal. The socket's open collector output drives the system bus' IRQn signal.

Level Mode

Level-Mode is the primary interrupt request type for level-sensitive-interrupt systems. Edge-sensitive interrupt systems can use Level-Mode if only one source at a time asserts IREQ#. The card requests an interrupt by forcing IREQ# low. The host ATA driver acknowledges the card's interrupt request; this is handled in various ways depending on the card's requirements. Upon interrupt-acknowledged, the card returns IREQ# to its de-asserted (high) level.

The socket must invert the PCMCIA IREQ# signal to produce IRQn in AT-bus systems that use Level-Mode interrupts. Additionally, the host must not share IRQn with other cards or devices unless it wire-ORs IRQn's sources.

Sharing Interrupts

When several devices share one IRQn, as in a wired-OR configuration, the system must differentiate between interrupt sources. Because of this complication, most AT systems don't use wired-OR configurations. They usually have one signal source per IRQn. A pull-up resistor on each IRQn maintains an "asserted" level when no interrupt is requested. Since these systems use positive-edge interrupts, IRQn must be in this condition when the host acknowledges an interrupt. An interrupting source pulls IRQn low and then releases it (it's pulled high by the pull-up) to generate the positive-edge interrupt.

Interrupt sources are effectively wire-ANDed together. Time-multiplexing is used to share the interrupt line. Each device has a three-state driver. Only one device at a time, when its three-state driver is enabled, can drive IRQn. This is typical in ISA AT implementations that share IRQ_{14} among several ATA drives. In these systems, PCMCIA's Pulse-Mode interrupts don't require the socket to recognize device-dependent protocols when driving the system's IRQn.

An ISA AT device asserts an interrupt by placing its output in a high-Z state. These devices usually use open drain (or open collector) outputs. The IRQn's pull-up resistor pulls the high-Z output, or outputs if shared, to a high level. If another device is driving IRQn low, it will remain low until all devices release IRQn. Only then will the interrupt controller recognize an interrupt. Devices that use three-state drivers must allow an interrupt controller to selectively enable interrupt outputs so only one device can assert an interrupt at any one time. The socket must control this time multiplexing.

Systems that use wired-OR interrupt lines can share them by using PCMCIA's Level-Mode interrupts. A pending interrupt request from any PCMCIA card or device will be routed to the desired IRQn where it is ORed with other requests. The interrupt controller will then request servicing from the host.

Interface Bus Timing

Several bus timings must be considered when using the flash drive in PCMCIA modes: reset, attribute (or CIS-tuple), common-memory, and I/O timing.

PCMCIA Reset Timing

The flash drive requires some initialization time after a power-on or hardware reset. This time is specified below.

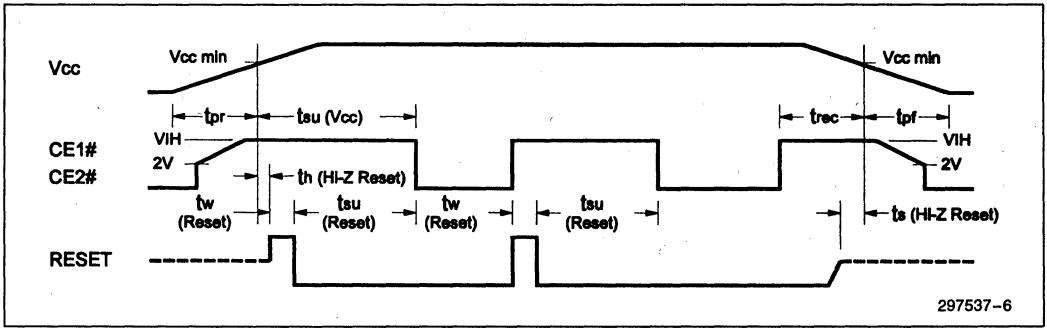


Figure 6. PCMCIA Mode Power-Up and Power-Down Reset Timing

Table 12. PCMCIA Mode Power-On and Reset Timing Parameters

Symbol	Parameter	Min	Max	Units
tsu (V_{CC})	CE# Setup Time from V_{CC}	20		ms
tsu (RESET)	CE# Setup Time from Reset	20		ms
trec (V_{CC})	CE# Recovery Time	0.001		ms
tpr(1)	V_{CC} Rise Time [10% to 0.9 (V_{CC} + 5%)]	0.1	300	ms
tpf(1)	V_{CC} Fall Time [0.9 (V_{CC} − 5%) to 10%]	3.0	300	ms
tw (Reset)	Reset pulse width	10		μs
th (Reset)	Reset high from V_{CC} min	1		ms
ts (Reset)	Reset high-Z from V_{CC} min	0		ms

NOTE:
1. tpr and tpf are "linear waveforms" during the 10% to 90% (or vice-versa) period. Even if the waveform is not a "linear waveform", its rise and fall times must meet this specification.

Attribute-Mode Timing

Attribute memory's access time is 300 ns at standard V_{CC} = 5V ±5% operating voltage. Figure 7 and Table 13 show detailed timing specifications.

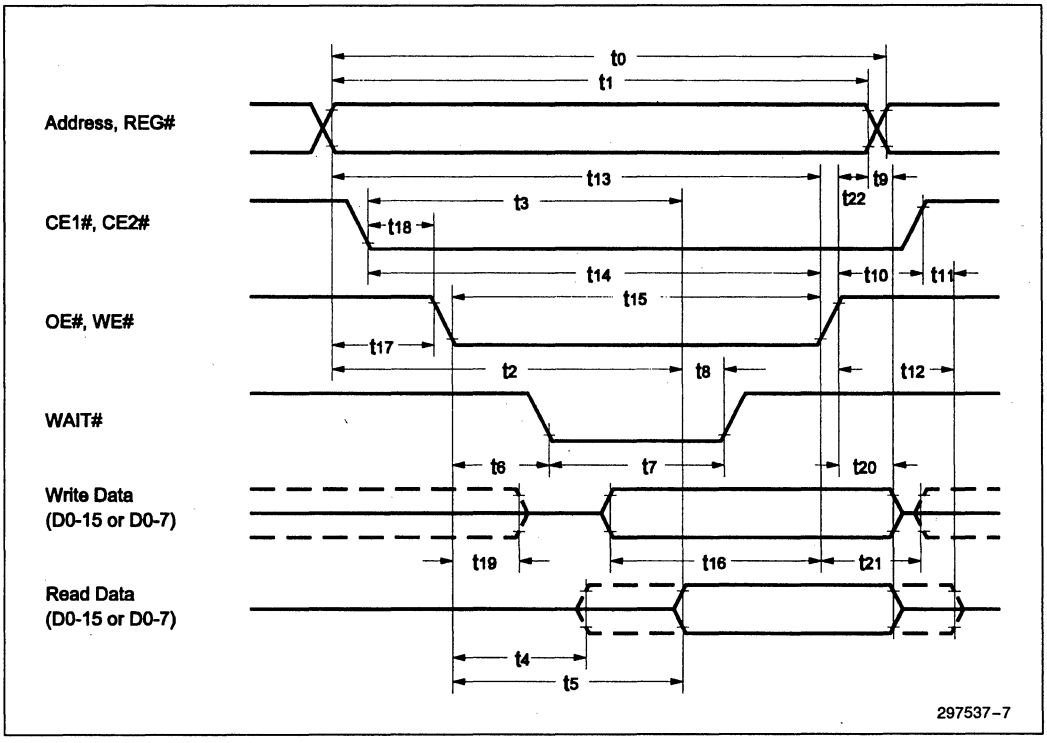


Figure 7. Attribute-Mode Timing Diagram

Table 13. Attribute-Mode Timing Parameters

Symbol	IEEE	Parameter	Min	Max	Units
READ					
t1	tAVAVR	Read Cycle Time	300		ns
t2	tAVQV	Address Access Time		300	ns
t3	tELQV	Card Enable Access Time		300	ns
t4	tGLQNZ	Output Enable to Data Change	5		ns
t5	tGLQV	Output Enable Access Time		150	ns
t6	tGLWTV	Wait Valid from Output Enable		35	ns
t7	tWTLWTH	Wait Pulse Width		12	μs
t8	tQVWTH	Data Setup for Wait Released	0		ns
t9	tAXQX	Data Valid after Address Change	0		ns
t10	tGHEH	Card Enable Hold Time	20		ns
t11	tEHQZ	Data Disable from Card Enable High		100	ns
t12	tGHQZ	Data Disable from Output Enable High		100	ns
WRITE					
t0	tAVAVW	Write Cycle Time	250		ns
t13	tAVWH	Address Valid to Write Enable High	180		ns
t14	tELWH	Card Enable to Write Enable High	180		ns
t15	tWLWH	Write Enable Pulse Width High	150		ns
t16	tDVWH	Data Valid to Write Enable High	80		ns
t17	tAVWL	Address Valid to Write Enable Low	30		ns
t18	tELWL	Card Enable Setup to Write Enable Low	0		ns
t19	tWLQZ	Write Enable to Previous Read-Data Disabled		100	ns
t20	tWHDX	Data Hold after Write Enable High	30		ns
t21	tWHQNZ	Data Driven after Write Enable High	5		ns
t22	tWHAX	Address Hold after Write Enable High	30		ns

PCMCIA-ATA Memory and I/O Timing

Once the flash drive's attribute memory has been read, the configuration option is written to the PCMCIA Configuration Option register. Attribute memory timing requirements are described above. Once configured for either memory- or an I/O-mode, card timings adhere to the ATA specification's programmed I/O (PIO) timing requirements shown in Figure 8 and Table 14.

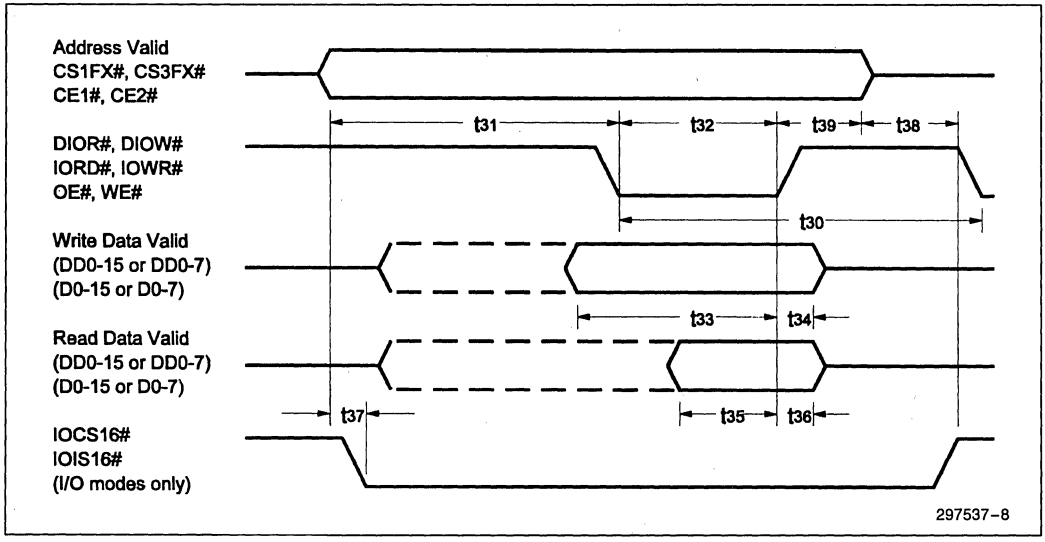


Figure 8. PCMCIA-ATA-Mode I/O and Memory Timing Diagram

Table 14. PCMCIA-ATA-Mode I/O and Memory Timing Parameters

Symbol	IEEE	Parameter	Min	Max	Units
t30	tIGLIGL	Cycle Time	383		ns
t31	tAVIGL	Address, Card Enable, to Read/Write Setup	50		ns
t32	tIGLIGH	Read/Write Pulse Width	125		ns
t33	tDVIGH	Data Setup to Write High	45		ns
t34	tIGHDX	Data Hold After Write High	20		ns
t35	tQVIGH	Data Setup to Read High	35		ns
t36	tIGHQX	Data Hold after Read High	0	5	ns
t37	tAVISL	Address or Card Enable Valid to IOCS16# Assertion	0	50	ns
t38	tAXISH	Address or Card Enable Valid to IOCS16# Negation		45	ns
t39	tIGHAX	Read/Write to Address or Card Enable Invalid	15		ns

Configuration Options Indexes

Table 15 summarizes the flash drive's four PCMCIA addressing modes. The Configuration Index codes are obtained from the tpce_indx entries in card configuration table entry tuples when the host reads the card's CIS. The host selects the addressing mode that best fits its resources by writing the Configuration Index to the Configuration Option register. The Configuration Tuple's tpcc_radr entry contains the Configuration Option register's address within the attribute memory space; this address is 200h for Intel Flash Drives. Figure 9 shows the card's addressing-mode decision process.

Table 15. PCMCIA Addressing Modes

Conf. Index	Address Mode	Address	Description
0	Memory	000-00F, 400-7FF	Memory Mapped
1	I/O	0-F	I/O, 16 Contiguous Addresses
2	I/O	1F0-1F7, 3F6-3F7	Primary I/O
3	I/O	170-177, 376-377	Secondary I/O

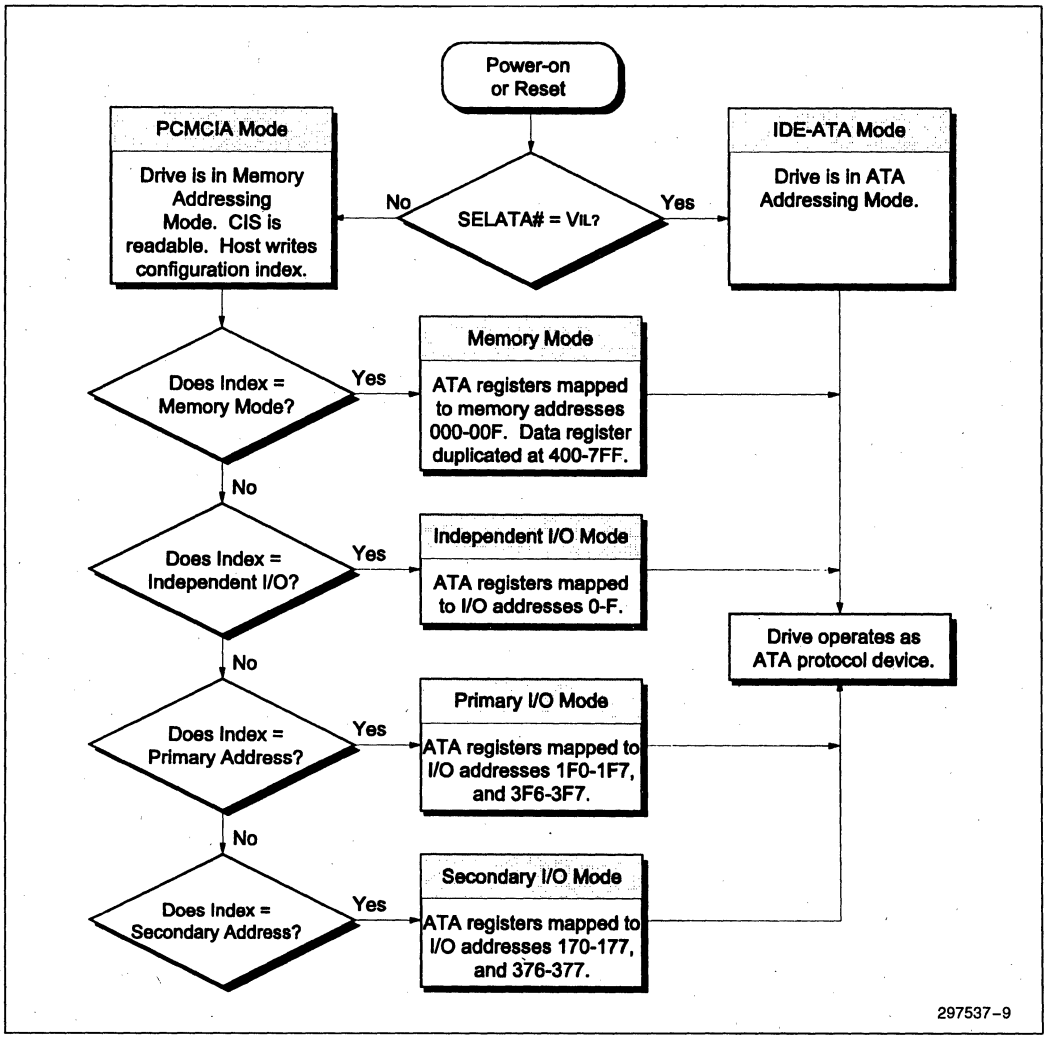


Figure 9. The Flash Drive Determines its Addressing Mode from the Configuration Index.

Card Configuration Registers

Table 16 summarizes the drive's four PCMCIA configuration registers. These read/write one-byte registers are located on even-byte attribute-plane addresses to ensure access in 8- and 16-bit systems. Available card status information allows arbitration between resources that share interrupts and status of memory-only-card pins 16, 33, 62, and 63.

Table 16. PCMCIA Card Configuration Registers Truth Table

Register	Add. Offset(1)	R/W	REG#	CE2#	CE1#	OE#	WE#	D15–8	D7–0
Configuration Option	200h	Read	L	H	L	L	H	Invalid	Option
		Write	L	H	L	H	L	Invalid	Option
Card Configuration and Status	202h(2)	Read	L	H	L	L	H	Invalid	Status
		Write	L	H	L	H	L	Invalid	Config.
Pin Replacement	204h(2)	Read	L	H	L	L	H	Invalid	Pin Status
		Write	L	H	L	H	L	Invalid	Pin Status
Socket and Copy	206h(2)	Read	L	H	L	L	H	Invalid	Socket ID
		Write	L	H	L	H	L	Invalid	Socket ID

NOTES:
1. The host obtains attribute-memory address offset from the Configuration Tuple's tpcc_radr field when it reads the drive's CIS.
2. Some engineering samples (marked ES) and early production units (marked iFD0xxP2SA B) do not support these registers.

Configuration Option Register

The host uses the read/write Configuration Option register to configure the drive for one of its four PCMCIA-ATA addressing modes, establish the interrupt signal mode, and issue a soft reset.

7	6	5	4	3	2	1	0
SRST	IRQLvl	Configuration Index					

SRST — Resets the card when 1. When 0 (default) after a hardware or software reset or power-on, the card is unconfigured. A configuration occurs when a valid configuration index is written to bits 0–5.

IRQLvl — Selects level mode interrupts when 1; pulse mode interrupts when 0 (default).

Conf IDX — The host chooses an option from the card's configuration table entry tuples and writes that option's Configuration Index number into this field. When zero (default), the memory-only interface is chosen; I/O accesses are disabled.

Configuration & Status Register

The Configuration and Status register contains card condition information.

7	6	5	4	3	2	1	0
Chng	SigChg	IOis8	0	Audio	PwrDn	Intr	0

Chng — The Change bit indicates that a Pin Replacement register bit was changed.

SigChg — The host sets/resets the Signal Changed bit to enable/disable a state-change signal from the status register. When set and the drive is configured for I/O, Chng controls pin 63 and is called the Changed Status signal. This bit should be 0 (BVD1/STSCHG# held high when configured for I/O) if no state change signal is desired.

IOis8 — Must be set if the host can only perform 8-bit I/O accesses.

Audio Audio is not supported.

PwrDn Setting PwrDn places the drive in sleep mode. Host-initiated ATA task-file-register commands can also invoke low-power modes.

Intr This bit represents the interrupt request's internal state. Its value is available whether or not interrupts are configured. It remains true until the initiating-interrupt request is serviced.

Pin Replacement Register

The Pin Replacement register provides card status information that is otherwise provided on memory-only interface pins 16, 33, 62, and 63. When written, bits 0-3 are masks for setting corresponding bits 4–7.

7	6	5	4	3	2	1	0
CBvd 1	CBvd 2	Crdy/ bsy#	CWP	RBvd 1	RBvd 2	Rrdy/ bsy#	RWP

CBvd1,2 Set when written, otherwise zero.

Crdy/ bsy# Set when Rrdy/bsy# changes state or when written by the host

CWP Zero unless set by the host.

RBvd1,2 Set unless cleared by the host.

Rrdy/ bsy# Internal ready/busy# state when I/O mode uses RDY/BSY# pin for interrupt.

RWP Zero, since the flash drive has no write protect switch, unless set by the host.

Socket and Copy Register

The Socket and Copy register allows the drive to distinguish between similar drives at the same address. The flash drive does not support this feature.

7	6	5	4	3	2	1	0
0	Copy Number			Socket Number			

Copy # The twin-card option is not supported.

Socket # The socket number is ignored.

CIS (Tuples)

The card information structure (CIS) contains information about the card's identity and manufacturer, its capabilities, functional modes, voltage, power, and addressing requirements, and other system resources needed for proper functionality. Attribute memory contains the CIS, or, as the PCMCIA specification calls it, the card metaformat.

Attribute memory can be thought of as a memory plane, similar to I/O and common memory. This plane keeps the CIS separate from the card's data media. It's accessed by asserting (low) the card's REG# signal. Then, byte accesses are performed by using the host processor's memory read and write functions (OE# and WE#).

The host reads the CIS by asserting REG#-low, CE1#-low, CE2#-high, OE#-low, WE#-high and even addresses ($A0 = V_{IL}$) starting at 0000h; odd address bytes are undefined. The device information tuple (code 01h) is at address 0.

In addition to device, interface, features, and manufacturer information, the CIS tuples describe the four PCMCIA-ATA configurations described earlier. The host picks one mode that fits its hardware and software requirements. It writes the mode's option index number to the Configuration Option register at card attribute address 200h.

All CIS tuples use the same format. The tuple's first byte is the tuple code (tpl_code). The second byte is a link (tpl_link) or offset to the next tuple code; it's the number of bytes, after the tpl_link byte, contained in the tuple. A zero in the link field means that the tuple body is empty. FFh in the link field means that the tuple is the last tuple in the list (Intel's Flash Drive does not use this convention; it uses the End of List tuple, tpl_end). In attribute memory, the link field describes the *number of bytes remaining* in the tuple, not the number of address locations remaining. The address offset to the next tuple is [2 * link field]. Third and successive tuple bytes describe information specific to the tuple. The tuple format is:

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
00	X	X	X	X	X	X	X	X	X	tpl_code	Tuple Code
02	X	X	X	X	X	X	X	X	X	tpl_link	Link to Next Tuple
...	X	X	X	X	X	X	X	X	X	tpl_xxx	Tuple-Specific Bytes

All multi-byte numeric data within tuples are in little-endian order; least-significant data byte is in the lower address. All character data is stored in the "natural order"; the field's first character is stored in the field's first byte.

Device Information Tuple
cistpl_device

The Device Information Tuple is always the first tuple and contains card-device information; speed, size, type, and common-memory address-space layout. Each field has two variable-length byte sequences (device ID and device size) and address-group characteristics within the memory space. The device ID describes the device type and the memory-block's access time.

The **device type** code describes the type of memory; code Dh specifies a function specific memory such as ATA.

WPS indicates whether the Write Protect Switch controls the device in this address range. When 0, the Write Protect Switch and WP-signal indicate whether the device is writable. When 1, the device is always writable.

The **device speed** field indicates the common memory access time. Codes are: 0 = null, 1 = 250 ns, 2 = 200 ns, 3 = 150 ns, 4 = 100 ns, 5-6 = reserved, 7 = use next byte for speed information. See the PCMCIA specification for extended speed codes.

The **device size** codes describe the number of address units (e.g. code zero indicates 1 unit, code one indicates 2 units, etc.) and the unit's size required in the common memory space.

Device Information Tuple

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
00	01	0	0	0	0	0	0	0	1	tpl_device	Common memory space
02	04	0	0	0	0	0	1	0	0	tpl_link	Link to next tuple
04	DF	Device Type Code 1	1	0	1	WPS 1	Dev Speed (See Next) 1	1	1	device_ info	device type; code D= function-specific memory address range. Includes memory mapped I/O registers and communication buffers not used as general memory. Write protect switch is not valid. Card writable. Device speed use next byte.
06	4A	Ext 0	Mantisa 4.0 x 1	0	0	1	Speed Exp 100 ns 0	1	0		extended speed byte; mantissa code 9= 4.0, exponent code 2= 100 ns, 4.0 x 100 ns = 400 ns t_{ACC}.
08	01	No. of addr units -1 0	0	0	0	0	size 0	0	1	device_ size	# of address units = 0+1=1; size code 1=2 Kbytes.
0A	FF	1	1	1	1	1	1	1	1	end	End device info field.

Configuration Tuple
cistpl_config

The CIS contains only one configuration tuple which describes a configurable card's general characteristics, I/O functions, or custom interfaces. The default is a memory-only interface.

The **size of fields** byte (tpcc_sz) contains three fields: tpcc_rfsz = area size (presently reserved, must be 0); tpcc_rmsz+1 = number of configuration register presence mask (tpcc_rmsk) field bytes; tpcc_rasz+1 = number of reg space base address field (tpcc_radr) bytes.

The **last entry index** byte (tpcc_last) contains the card configuration table's last-described configuration index number. Once encountered, the host knows that it has processed all valid configurations.

The 1-to-4 byte **REG space base address** (tpcc_radr) describes the configuration register's even-byte base address in attribute memory. This, and the configuration register's presence-mask field, indicate which card registers are present and their register-space location. High-order address bits not present in the field are zero.

The **configuration register presence mask field** (tpcc_rmsk) bits represent the corresponding configuration register's presence (1) or absence (0). The least-significant bit represents the Configuration Option register; other bits represent, in order, registers listed in Table 16. This, and the REG space base address bytes, indicate which card registers are present and their register-space location. Each configuration register address is: Base address + (register number * 2).

Configuration Tuple

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
0C	1A	0	0	0	1	1	0	1	0	tpl_config	Configuration tuple code
0E	05	0	0	0	0	0	1	0	1	tpl_link	
10	01	tpcc_rfsz 0	0	0	tpcc_rmsz 0	0	0	tpcc_rasz 0	1	tpcc_sz	size of fields; tpcc_rfsz = zero (reserved), tpcc_rmsk=tpcc_rmsz+1=1, tpcc_radr=tpcc_rasz+1=2.
12	03	Reserved 0	0	Index number of final card config table entry 0	0	0	0	1	1	tpcc_last	last card configuration table index number=3.
14	00	0	0	0	0	0	0	0	0	tpcc_radr	configuration register base address = 200h.
16	02	0	0	0	0	0	0	1	0		
18	0F	0	0	0	0	1(1)	1(1)	1(1)	1	tpcc_rmsk	All config registers present.

NOTE:
1. Engineering samples (marked ES) and early production units (marked iFD0xxP2SA B) do not support the Card Configuration and Status, Pin Replacement, and Socket and Copy registers.

Card Configuration Table Entry Tuples

cistpl_cftable_entry

The system uses the Card Configuration Table Entry tuples to determine the card's configurable characteristics. The system may configure the card to an allowable configuration or leave it unconfigured (to the default memory mode). It may use the configuration to gracefully reject the card if it is incompatible with the system's recourses. It can also describe a PC card's voltage and current requirements.

The **configuration table index** byte contains the value written to the Card Configuration register to enable this tuple's described configuration. It also contains bits that indicate that an interface description byte follows and that this tuple defines default conditions for following configuration entry tuples. When the default bit is set, the configuration table entry provides default values for following card configuration table entries. When not set, the tuple provides default values for this configuration only.

The **interface description field** indicates the configuration's memory-only or memory-I/O card interface type. It also indicates the card's battery voltage detect, write protect, ready/busy, and wait signal system requirements.

The **features selection byte** indicates the table's descriptive field presence (1 in the field bit). The fields' order is determined by the entry's bit order (bit 0=LSB, bit 7=MSB). Descriptions may include: V_{CC}/V_{PP} voltages, timing, I/O addresses and ranges, interrupt request number recommendations, memory space requirement, and miscellaneous field presence. Intel's flash drive contains descriptions for nominal V_{CC}, I/O, IRQ, memory, and miscellaneous fields only.

Card Configuration Table Entry Tuple—Memory Space Definition

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
1A	1B	0	0	0	1	1	0	1	1	cftbl_entry	card config table entry
1C	08	0	0	0	0	1	0	0	0	tpl_link	
1E	C0	I/F 1	Default 1	Configuration Entry Index Number 0	0	0	0	0	0	tpce_indx	configuration index byte; I/F byte follows, default, config table index = 00.
20	C0	Wait 1	Ry/By 1	WP 0	BVD 0	I/F type 0	0	0	0	tpce_if	interface description field; wait req'd, rdy/bsy active, no write-protect switch, no battery voltage detect, I/F code 0 = memory.
22	A1	misc 1	mem space 0	1	IRQ 0	I/O 0	Timing 0	V_{PP} & V_{CC} 0	1	tpce_fs	features selection byte; misc byte present, mem space = 1 + 1 = 2 bytes, no IRQ recommendation, no I/O defined, no timing described, V_{CC} described.
24	01	resv'd 0	PD I 0	Peak I 0	Avg I 0	Stat I 0	Max V 0	Min V 0	Nom V 1	tpce_pd	power-description; nominal V_{CC} described.
26	55	Ext 0	mantissa = 5.0 x 1	0	1	0	exponent = 1V 1	0	1		nominal V_{CC} description mantissa code A = 5, exponent code 5 = 1V, V_{CC} nom = 5V ±5%.
28 2A	08 00	0 0	0 0	0 0	0 0	1 0	0 0	0 0	0 0	tpce_ms	memory space description; map 8 x 256 = 2048 bytes.
2C	20	Ext 0	Resv'd 0	PwrDn 1	Rdonly 0	Audio 0	Max Twin Cards 0	0	0	tpce_mi	miscellaneous field; power-down supported, not read-only, audio not supported, twin cards not supported.

Card Configuration Table Entry Tuple—I/O space Definition

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
2E	1B	0	0	0	1	1	0	1	1	cftbl_entry	card config table entry
30	0A	0	0	0	0	1	0	1	0	tpl_link	
32	C1	I/F 1	Default 1	Configuration Entry Number 0	0	0	0	0	1	tpce_indx	configuration index byte; I/F byte follows, default, config table index = 01.
34	41	Wait 0	Ry/By 1	WP 0	BVD 0	I/F type 0	0	0	1	tpce_if	interface description field; wait not supported, ready/busy active, no write-protect switch, no battery voltage detect, I/F code 1 = I/O.
36	99	misc 1	mem space 0	0	IRQ 1	I/O 1	Timing 0	V_{PP} & V_{CC} 0	1	tpce_fs	features selection byte; misc present, IRQ, I/O, V_{CC} described.
38	01	resv'd 0	PD I 0	Peak I 0	Avg I 0	Static I 0	Max V 0	Min V 0	Nom V 1	tpce_pd	power-description; nominal V_{CC} described.
3A	55	Ext 0	5.0 x 1	0	1	0	1 Volt 1	0	1		nominal V_{CC} description mantissa code A = 5, exponent code 5 = 1 Volt, V_{CC} nom = 5V ±5%.
3C	64	ext 0	16-bit 1	8-bit 1	I/O address lines 0	0	1	0	0	tpce_io	I/O space description; 8 or 16-bit bus supported, 4 I/O lines (A_{0-3}) required.
3E	F0	Share 1	Pulse 1	Level 1	Mask 1	IRQ Number = 0 0	0	0	0	tpce_ir	interrupt request; sharing IRQs is OK, pulse & level IRQ OK, IRQ mask follows.
40	FF	IRQ7 1	IRQ6 1	IRQ5 1	IRQ4 1	IRQ3 1	IRQ2 1	IRQ1 1	IRQ0 1		Any IRQ_{0-15} OK.
42	FF	IRQ15 1	IRQ14 1	IRQ13 1	IRQ12 1	IRQ11 1	IRQ10 1	IRQ9 1	IRQ8 1		
44	20	Ext 0	Resv'd 0	PwrDn 1	Rd 0	Audio 0	Max Twin Cards 0	0	0	tpce_mi	miscellaneous field; power-down supported, not read-only, audio not supported, twin cards not supported.

7

Card Configuration Table Entry Tuple—Primary Drive-Address Definition (1F0–1F7, 3F6–3F7)

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
46	1B	0	0	0	1	1	0	1	1	cftbl_entry	card config table entry
48	0C	0	0	0	0	1	1	0	0	tpl_link	
4A	82	I/F 1	Default 0	Configuration Entry Number 0	0	0	0	1	0	tpce_indx	configuration index byte; I/F byte follows, no default, config table index = 02.
4C	41	Wait 0	Ry/By 1	WP 0	BVD 0	I/F type 0	0	0	1	tpce_if	interface description field; wait not supported, ready/busy active, no write-protect switch, no battery voltage detect, I/F code 1 = I/O.
4E	18	misc 0	mem space 0	0	IRQ 1	I/O 1	Timing 0	V_{PP}&V_{CC} 0	0	tpce_fs	features selection byte; IRQ and I/O described.
50	EA	Ext 1	16-bit 1	8-bit 1	I/O address lines 0	1	0	1	0	tpce_io	I/O space description; extended bytes present, 8 or 16-bit bus supported, 10 I/O lines (A_{0-9}) required.
52	61	length = 1 byte 0	1	add = 2 bytes 1	0	I/O addr ranges = 1 + 1 = 2 0	0	0	1		# of I/O address range fields that follow; length = 1 byte, address = 2 bytes, ranges described = 2 bytes.
54	F0	1	1	1	1	0	0	0	0		first I/O range;
56	01	0	0	0	0	0	0	0	1		I/O start address = 1F0h.
58	07	0	0	0	0	0	1	1	1		I/O block length = 7 + 1 = 8.
5A	F6	1	1	1	1	0	1	1	0		second I/O range;
5C	03	0	0	0	0	0	0	1	1		I/O start address = 3F6h.
5E	01	0	0	0	0	0	0	0	1		I/O block length = 1 + 1 = 2.
60	EE	Share 1	Pulse 1	Level 1	Mask 0	IRQ Number 1	1	1	0	tpce_ir	interrupt request; sharing IRQs is OK, pulse & level IRQ OK, no IRQ mask, recommend IRQ_{14}.

Card Configuration Table Entry Tuple—Secondary Drive-Address Definition (170–177, 376–377)

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
62	1B	0	0	0	1	1	0	1	1	cftbl_entry	card config table entry
64	0C	0	0	0	0	1	1	0	0	tpl_link	
66	83	I/F 1	Default 0	Configuration Entry Number 0	0	0	0	1	1	tpce_indx	configuration index byte; I/F byte follows, no default, config table index= 03.
68	41	Wait 0	Ry/By 1	WP 0	BVD 0	I/F type 0	0	0	1	tpce_if	interface description field; wait not supported, ready/busy active, no write-protect switch, no battery voltage detect, I/F code 1= I/O.
6A	18	misc 0	mem space 0	0	IRQ 1	I/O 1	Timing 0	V_{PP} & V_{CC} 0	0	tpce_fs	features selection byte; IRQ and I/O described.
6C	EA	ext 1	16-bit 1	8-bit 1	I/O address lines 0	1	0	1	0	tpce_io	I/O space description; extended bytes present, 8 or 16-bit bus supported, 10 I/O lines (A_{0-9}) required.
6E	61	length=1 byte 0	1	add=2 bytes 1	0	I/O addr ranges=1+1=2 0	0	0	1		# of I/O address range fields that follow; length= 1 byte, address= 2 bytes, ranges described= 2 bytes.
70 72	70 01	0 0	1 0	1 0	1 0	0 0	0 0	0 0	0 1		first I/O range; I/O start address= 170h.
74	07	0	0	0	0	0	1	1	1		I/O block length = 7+1= 8.
76 78	76 03	0 0	1 0	1 0	1 0	0 0	1 0	1 1	0 1		second I/O range; I/O start address = 376h.
7A	01	0	0	0	0	0	0	0	1		I/O block length = 1+1= 2.
7C	EE	Share 1	Pulse 1	Level 1	Mask 0	IRQ Number 1	1	1	0	tpce_ir	interrupt request; sharing IRQs is OK, pulse & level IRQ OK, no IRQ mask, recommend IRQ_{14}.

Function ID Tuple for Disk Function

cistpl_funcid

The Function ID tuple specifies a disk-device type card. Two Function Extension tuples follow which further specify the PC Card ATA disk interface protocol.

Function ID Tuple for Disk Function

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
7E	21	0	0	1	0	0	0	0	1	tpl_funce	disk drive function ID tuple
80	02	0	0	0	0	0	0	1	0	tpl_link	
82	04	0	0	0	0	0	1	0	0	tplfid_func	04= fixed disk function.
84	01	0	0	0	0	0	0	0	1	tplfid_ sysinit	system init flags; 01= host POST must configure card.

Disk Interface Function Extension Tuple

cistpl_funce

This tuple always follows the Function ID Tuple for Disk Function. It specifies the disk function device interface protocol. All byte values are pre-defined by PCMCIA.

Disk Device Interface Function Extension Tuple

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
86	22	0	0	1	0	0	0	1	0	tpl_funce	disk function extension
88	02	0	0	0	0	0	0	1	0	tpl_link	
8A	01	0	0	0	0	0	0	0	1	tplfe_type	disk I/F extension type.
8C	01	0	0	0	0	0	0	0	1	tplfd_data	PC card ATA interface.

ATA Features Function Extension Tuple

cistpl_funce

This tuple follows the Function ID tuple for Disk Function and specifies the PC Card ATA disk-function related features described in the Function ID tuple. The first three bytes are pre-defined by PCMCIA.

The **PC Card ATA Features** byte 1 (tplfe_data) describes V_{PP} voltage requirements (0 = not required), rotating (0) or silicon (1) technology, and model/serial number uniqueness (1).

The **PC Card ATA Features** byte 2 (tplfe_data) describes low-power options, floppy-disk compatibility at I/O addresses 3F7h and 377h, disk-drive index bit emulation, and IOIS16# use in twin-card configurations.

PC Card ATA Features Function Extension Tuple

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
8E	22	0	0	1	0	0	0	1	0	tpl_funce	disk function extension
90	03	0	0	0	0	0	0	1	1	tpl_link	
92	02	0	0	0	0	0	0	1	0	tplfe_type	disk I/F extension type.
94	0C	Reserved (must be 0's) 0	0	0	0	drv ID 1	Silic'n 1	V_{PP} Req'd 0	0	tplfe_data	card ATA features byte 1; unique model/SN ID, rotating device emulation, no V_{PP} voltage required.
96	0F	Resv'd 0	I/O16 0	Index 0	Floppy 0	autopw 1	Idle 1	stndby 1	Sleep 1	tplfe_data	card ATA features byte 2; twin-cards not supported, no disk-drive index bit, primary/secondary I/O modes include 3F7h/377h, supports auto-power-down, idle, standby, and sleep low-power modes.

ATA JEDEC Identifier Tuple
cistpl_jedec_c

A PC Card ATA JEDEC Identifier tuple corresponds to each device described in the Device Information tuple. Byte 3, the JEDEC manufacturer ID, has a pre-defined ID of 95 decimal (5Fh); bit-7 is set for parity making the code DFh. Byte 4, the ATA device ID, also indicates the Vpp requirement; 1 = no Vpp required (Intel Flash Drive default), 2 = Vpp required for media modification only, 4 = Vpp required for all media access, 8 = continuous Vpp required.

PC Card ATA JEDEC Identifier Tuple

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
98	18	0	0	0	1	1	0	0	0	tpl_jedec	ATA JEDEC ID
9A	02	0	0	0	0	0	0	1	0	tpl_link	
9C	DF	1	1	0	1	1	1	1	1	JedecID 1	manufacturer ATA ID = DFh
9E	01	0	0	0	0	0	0	0	1	JedecID 2	ATA device ID = 1, no V_{PP}.

Manufacturer Identification Tuple
cistpl_manfid

The Manufacturer Identification tuple describes PC card manufacturer's ID code (the manufacturer's JEDEC-assigned code; Intel's code = 89h) and the card's part number.

Manufacturer Identification Tuple

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
A0	20	0	0	1	0	0	0	0	0	tpl_manfid	manufacturers ID tuple
A2	04	0	0	0	0	0	1	0	0	tpl_link	
A4	89	1	0	0	0	1	0	0	1	tplmid_	Intel's manufacturer code
A6	00	0	0	0	0	0	0	0	0	manf	= 89h
A8	AF	1	0	1	0	1	1	1	1	tplmid_	part number = AFh.
AA	00	0	0	0	0	0	0	0	0	card	

Level 1 Version/Product Information Tuple
cistpl_vers_1

This tuple contains PCMCIA level-1-version compliance and card manufacturer information. The first four bytes are pre-defined by PCMCIA. Remaining bytes contain ASCII characters that describe the manufacturer's name, product information string, and an additional product version string. 00h terminates each string. The last string indicates the end of device information with code FFh.

The Level 1 Version/Product Information tuple is sometimes used to determine whether the card is compatible with the host system. The benefit of this approach is that systems can recognize a card by matching this tuple's contents with a system lookup table. This expedites card recognition and greatly simplifies the tuple-parsing process. Disadvantages are that these systems only recognize cards listed in their lookup table, they won't work with compatible cards from other manufacturers, won't work with future card products, and can't implement special card features. Intel recommends that systems properly parse all tuples, especially the Configuration Table Entry tuples, to determine compatibility. The host should always configure the card by writing a valid configuration index to the Configuration Option register.

Level 1 Version/Product Information Tuple

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
AC	15	0	0	0	1	0	1	0	1	tpl_vers_1	Version 1 compliance
AE	1C	0	0	0	1	1	1	0	0	tpl_link	
B0	04	0	0	0	0	0	1	0	0	tpllv1_maj	major version number = 04
B2	01	0	0	0	0	0	0	0	1	tpllv1_min	minor version number = 01
B4	49	0	1	0	0	1	0	0	1	I	tpllv1_info
B6	6E	0	1	1	0	1	1	1	0	n	manufacturer's information
B8	74	0	1	1	1	0	1	0	0	t	
BA	65	0	1	1	0	0	1	0	1	e	
BC	6C	0	1	1	0	1	1	0	0	l	
BE	00	0	0	0	0	0	0	0	0		end manufacturer info
C0	46	0	1	0	0	0	1	1	0	F	product information string
C2	6C	0	1	1	0	1	1	0	0	l	
C4	61	0	1	1	0	0	0	0	1	a	
C6	73	0	1	1	1	0	0	1	1	s	
C8	68	0	1	1	0	1	0	0	0	h	
CA	44	0	1	0	0	0	1	0	0	D	
CC	72	0	1	1	1	0	0	1	0	r	
CE	69	0	1	1	0	1	0	0	1	i	
D0	76	0	1	1	1	0	1	1	0	v	
D2	65	0	1	1	0	0	1	0	1	e	
D4	00	0	0	0	0	0	0	0	0		end of product info string
D6	69	0	1	1	0	1	0	0	1	i	additional product info
D8	46	0	1	0	0	0	1	1	0	F	
DA	44	0	1	0	0	0	1	0	0	D	
DC	50	0	1	0	1	0	0	0	0	P	
DE	32	0	0	1	1	0	0	1	0	2	
E0	53	0	1	0	1	0	0	1	1	S	
E2	41	0	1	0	0	0	0	0	1	A	
E4	00	0	0	0	0	0	0	0	0		end additional product info
E6	FF	1	1	1	1	1	1	1	1		end device info

No-link Control Tuple

cistpl_no-link

The No-link Control tuple prevents tuple-parsing software from executing past the tuple chain (e.g. no long-link operations). This tuple contains two bytes: the tuple code (14h) and a zero-length tuple-link. An End-of-List tuple (code FFh) follows which marks the end of the tuple chain.

No-link Control Tuple (End of Tuples Marker)

Byte	Val	D7	D6	D5	D4	D3	D2	D1	D0	Code	Comments
E8	14	0	0	0	1	0	1	0	0	tpl_no_lnk	no-link control tuple
EA	00	0	0	0	0	0	0	0	0	tpl_link	zero-length-link
EC	FF	1	1	1	1	1	1	1	1	tpl_end	end-of-list tuple

Card Configuration

As the host parses the CIS's tuple chain, it looks for a memory or I/O configuration that matches its resources. The card's configurable options are described in the Configuration Table Entry tuples. The host writes the configuration's index number (obtained from the Configuration Table Entry tuple's tpce_indx field) to the Configuration Option register (its attribute-memory address is found in the Configuration tuple's tpcc_radr field). Table 15 summarizes the flash drive's configurable options. Although the card resets to and powers up in memory mode, configuration 00, the host should always select the proper mode by writing the Configuration Option register. Once configured, all card accesses should be via the ATA registers. Although the PCMCIA registers are available, ATA control and status register functions are more easily implemented and contain more reliable data than PCMCIA registers.

Twin Card Support

Intel's PCMCIA flash drive does not support twin-card operation. This option allows up to eight similar cards to occupy the same I/O port address. This feature has very little benefit and, as such, is not supported by BIOS ISVs. The I/O Mode Adapters section describes a more universal method for supporting similarly-configured multiple cards in a system.

Intel 82365SL and Compatible Chips

Intel's i82365SL PCMCIA (PCIC) adapter can access two sockets and can be cascaded to access up to eight sockets. Five memory and two I/O windows can map card addresses into the systems memory and I/O spaces. Any of 10 IRQs can route card interrupts to system interrupt inputs.

The i82365SL can handle Primary and Secondary drive I/O address but can't place bit-7 of I/O addresses 3F7h and 377h into a high-Z state to prevent conflict with a floppy-disk controller's Disk Change bit. There is also no means of dealing with two drives in a master/slave configuration at the same I/O address.

In memory mode, the i82365SL will not allow card writes (e.g. to the card's configuration and ATA task-file registers) if the card's write Protect/IOIS16# signal is high. Therefore, the flash drive asserts-low WP/IOIS16# immediately after reset until the card is configured into one of its I/O modes.

The controller only supports level-mode interrupts, not pulse-mode. Its IRQ outputs are high-to-low driven or high-Z. Therefore, IRQs can't be shared with other devices (e.g. IDE-ATA or other PCMCIA controllers). Only cards controlled by the same i82365SL should share an IRQ.

The i82365SL does not support INPACK#. Cards assert INPACK# whenever a valid I/O address is accessed. The system's sockets use INPACK# to enable input data-bus buffers. External logic should combine the card's CE1#, CE2#, IORD#, OE#, and INPACK# to enable the data buffer's outputs and direction control.

New i82365SL implementations and compatible devices may treat these conditions differently. Consult manufacturers data sheets for specific details.

ATA REGISTERS

Drive communication is via I/O registers that route input/output information to/from registers depending on host-supplied address and control signals.

Command block registers receive host commands or post drive status.

Control Block registers control the drive and post status via the Alternate Status register.

Table 17 lists these registers and their address-offsets.

Table 17. ATA Registers

CS1FX# (CE1#)	CS3FX# (CE2#)	DA_2	DA_1	DA_0	Offset	READ (DIOR#)	WRITE (DIOW#)
Command Block Registers							
V_{IL} (0)	V_{IL} (0)	X	X	X	X	Data (DD_{0-15})	Data (DD_{0-15})
V_{IL} (0)	V_{IH} (1)	V_{IL} (0)	V_{IL} (0)	V_{IL} (0)	0	Data (DD_{0-15})	Data (DD_{0-15})
V_{IL} (0)	V_{IH} (1)	V_{IL} (0)	V_{IL} (0)	V_{IH} (1)	1	Error	Features
V_{IL} (0)	V_{IH} (1)	V_{IL} (0)	V_{IH} (1)	V_{IL} (0)	2	Sector Count	Sector Count
V_{IL} (0)	V_{IH} (1)	V_{IL} (0)	V_{IH} (1)	V_{IH} (1)	3	Sector Number (LBA_{0-7})	Sector Number (LBA_{0-7})
V_{IL} (0)	V_{IH} (1)	V_{IH} (1)	V_{IL} (0)	V_{IL} (0)	4	Cylinder Low (LBA_{8-15})	Cylinder Low (LBA_{8-15})
V_{IL} (0)	V_{IH} (1)	V_{IH} (1)	V_{IL} (0)	V_{IH} (1)	5	Cylinder High (LBA_{16-23})	Cylinder High (LBA_{16-23})
V_{IL} (0)	V_{IH} (1)	V_{IH} (1)	V_{IH} (1)	V_{IL} (0)	6	Drive Head (LBA_{24-27})	Drive Head (LBA_{24-27})
V_{IL} (0)	V_{IH} (1)	V_{IH} (1)	V_{IH} (1)	V_{IH} (1)	7	Status	Command
Control Block Registers							
V_{IH}	V_{IH}	X	X	X	X	Data bus high-Z	Not Used
V_{IH}	V_{IL}	0	X	X	0-3	Data bus high-Z	Not Used
V_{IH}	V_{IL}	1	0	X	4-5	Data bus high-Z	Not Used
V_{IH}	V_{IL}	1	1	0	6	Alternate Status	Drive Control
V_{IH}	V_{IL}	1	1	1	7	Drive Address	Not Used

Data Register (Read/Write)

CS1FX# = 0, Offset = 0

Data block PIO (programmed I/O) transfers between the drive's data buffer and the host and format track command sector-information transfers use this 16-bit register.

F	E	D	C	B	A	9	8
D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
7	6	5	4	3	2	1	0

Error Register (Read)
CS1FX# = 0, Offset = 1

The Error register contains last drive-command-executed status or a diagnostic code. The Error register's contents are valid when the Status register's ERR is 1 at any command completion, except *execute drive diagnostic.* It contains a diagnostic code (see table 18) following power-on, reset, or *execute drive diagnostic* command.

7	6	5	4	3	2	1	0
BBK	UNC	MC	IDNF	MCR	ABRT	T0NF	AMNF

BBK Bad Block mark detected in the requested sector's ID field

UNC UNCorrectable data error encountered

MC Removable media access ability has Changed—not supported

IDNF Requested sector's ID-field Not Found

MCR Media Change Request indicates that the removable-media drive's latch has changed, indicating that the user wishes to remove the media—not supported

ABRT Drive status error or ABorted invalid command

T0NF Track 0 Not Found during a *recalibrate* command

AMNF Address Mark Not Found after finding the correct ID field.

Unused bits are zero.

Table 18: Diagnostic Codes

Code	Description
01h	No error detected
02h	Formatter device error
03h	Sector buffer error

Code	Description
04h	ECC circuitry error
05h	Controlling microprocessor error
8xh	Drive 1 failed (not used)

Set Feature Register (Write)
CS1FX# = 0, Offset = 1

This register enables drive-specific features. See the *set features* or *Intel get/set features* command descriptions.

Sector Count Register (Read/Write)
CS1FX# = 0, Offset = 2

This register contains the number of data sectors to be transferred during a read or write operation. A zero register value specifies 256 sectors. The command was successful if this register is zero at command completion. If the request is not completed, the register contains the number of sectors left to be transferred.

Some commands (e.g. *initialize drive parameters* or *format track)* may redefine this register's contents.

7	6	5	4	3	2	1	0
SC7	SC6	SC5	SC4	SC3	SC2	SC1	SC0

Sector Number Register (Read/Write)

CS1FX# = 0, Offset = 3

In CHS (cylinder, head, sector) mode, this register contains the subsequent command's starting sector number, which can be from 1 to the maximum number of sectors per track. See the command descriptions for register contents at command completion (whether successful or unsuccessful).

In LBA (logical block address) mode, this register contains LBA bits 0–7, which are updated at command completion.

7	6	5	4	3	2	1	0
SN_7 LBA_7	SN_6 LBA_6	SN_5 LBA_5	SN_4 LBA_4	SN_3 LBA_3	SN_2 LBA_2	SN_1 LBA_1	SN_0 LBA_0

Cylinder Low Register (Read/Write)

CS1FX# = 0, Offset = 4

In CHS mode, this register contains the cylinder number's low-8 bits at disk access start or command completion.

In LBA mode, this register contains LBA bits 8–15 and reflects their status at command completion.

7	6	5	4	3	2	1	0
CL_7 LBA_{15}	CL_6 LBA_{14}	CL_5 LBA_{13}	CL_4 LBA_{12}	CL_3 LBA_{11}	CL_2 LBA_{10}	CL_1 LBA_9	CL_0 LBA_8

Cylinder High Register (Read/Write)

CS1FX# = 0, Offset = 5

In CHS mode, this register contains the cylinder number's high-8 bits at disk access start or command completion.

In LBA mode, this register contains LBA bits 16–23 and reflects their status at command completion.

7	6	5	4	3	2	1	0
CH_7 LBA_{23}	CH_6 LBA_{22}	CH_5 LBA_{21}	CH_4 LBA_{20}	CH_3 LBA_{19}	CH_2 LBA_{18}	CH_1 LBA_{17}	CH_0 LBA_{16}

Drive/Head Register (Read/Write)

CS1FX# = 0, Offset = 6

This register contains drive and head numbers (heads minus 1, when executing *initialize drive parameters* command).

7	6	5	4	3	2	1	0
1	L	1	DRV	HS_3 LBA_{27}	HS_2 LBA_{26}	HS_1 LBA_{25}	HS_0 LBA_{24}

L — Address mode select. 0 = CHS (cylinder, head, sector) mode; 1 = LBA (logical block address) mode

DRV — Binary drive select number; DRV = 0 selects drive 0, DRV = 1 is ignored (on HDDs this bit selects drive 1)

HS_{3-0} — If L = 0: HS_{3-0} is the selected head's binary coded address (e.g. if HS_{3-0} is 0011b, head 3 is selected). At command completion, these bits reflect the currently selected head

If L = 1: HS_{3-0} contain LBA bits 24–27 and reflect their status at command completion.

Status Register (Read)
CS1FX# = 0, Offset = 7

This register contains drive status, which is updated at each command's completion. When BSY is cleared, register bits are valid within 400 ns. If set, no other Status register bits are valid. A host read, while interrupt is pending, constitutes interrupt acknowledge which clears any pending interrupt.

7	6	5	4	3	2	1	0
BSY	DRDY	DWF	DSC	DRQ	CORR	IDX	ERR

BSY The drive can access command block registers (the host should not access them) when set; any command block register read returns the Status register contents. The drive sets BSY:

within 400 ns after RESET# negation or after setting the Device Control register's SRST. Following reset, the drive sets BSY for up to 31 seconds

within 400 ns of host command: *read, read long, read buffer, seek, recalibrate, read verify, initialize drive parameters, identify drive, execute drive diagnostic*

within 5 µs after a 512 byte *write, format track, write buffer,* or data-plus-ECC *write long* command

DRDY Drive Ready indicates that the drive can respond to a command. On error, DRDY changes only after the host reads the Status register; the bit again indicates drive readiness. Power-on clears this bit; it remains clear until the drive can accept a command

DWF Drive Write Fault status. On error, it changes only after the host reads the Status register; the bit again indicates the current write fault status

DSC Drive Seek Complete indicates that the requested sector was found. On error, it changes only after a host Status register read; the bit again indicates the current seek complete status

DRQ Data Request indicates that the drive can transfer a data word or byte

CORR Indicates that a data error was corrected; transfer is not terminated

IDX Index is not used

ERR Error occurred during the previous command's execution. The Error register indicates the cause.

Command Register (Write)
CS1FX# = 0, Offset = 7

Execution commences when this register receives an executable host command with valid parameters.

Alternate Status Register (Read)
CS3FX# = 0, Offset = 6

This register contains command block Status information (see Status register). Unlike the Status register, reading this register does not acknowledge or clear an interrupt.

7	6	5	4	3	2	1	0
BSY	DRDY	DWF	DSC	DRQ	CORR	IDX	ERR

Drive Control Register (Write)

CS3FX# = 0, Offset = 6

This register contains two control bits:

7	6	5	4	3	2	1	0
—	—	—	—	—	SRST	IEN#	—

SRST Resets the drive when set, enables it when cleared.

IEN# Controls Interrupt Enable to the host (three-state pin INTRQ) when the drive is selected. When inactive (1) or drive not selected, INTRQ is high-Z regardless of a pending interrupt.

Drive Address Register (Read)

CS3FX# = 0, Offset = 7

This register reflects the drive and its heads.

7	6	5	4	3	2	1	0
High-Z	WTG#	HS3#	HS2#	HS1#	HS0#	DS1#	DS0#

High-Z For floppy drive address-space compatibility, bit 7 is not driven (it is high-Z) when this register's address is accessed in IDE-ATA mode. This allows a floppy-drive's disk-change bit to provide bit 7 data

WTG# Write Gate bit is active during in-progress disk writes (not supported)

HS_{3_0}# Ones complement of currently selected head number. For example, head 3 is selected if HS_{3_0} is 1100b

DS1# Drive Select 1 is low when drive 1 is selected and active (not supported)

DS0# Drive Select 0 is low when drive 0 is selected and active.

Addressing Mode Descriptions

The flash drive, on a command by command basis, can operate in either CHS or LBA addressing modes. Identify Drive Information tells the host whether the drive supports LBA mode.

The host selects LBA mode via the Drive/Head register. Sector Number, Cylinder Low, Cylinder High, and Drive/Head register bits HS_{3_0} contain the zero-based LBA. The drive's sectors are linearly mapped with:

LBA_0 = Cylinder 0, head 0, sector 1.

Regardless of the translation mode, a sector's LBA address does not change.

LBA = [(Cylinder * no of heads + heads) * sectors/track] + (Sector-1).

ATA COMMANDS

Writing Command register codes and block-register parameters issues commands. The drive responds and executes a new command by aborting an in-progress command, even if it can be completed.

Table 19. Command Register Codes

Command Code (hex)	Command Name	Features Register	Sector Count	Sector Number	Cylinder Registers	Drive/ Head
10-1F	Recalibrate					Drv
20	Read Sector(s) (w/ retry)		Valid	Valid	Valid	Drv/Hd
21	Read Sector(s) (w/o retry)		Valid	Valid	Valid	Drv/Hd
22	Read Long (w/ retry)		Valid	Valid	Valid	Drv/Hd
23	Read Long (w/o retry)		Valid	Valid	Valid	Drv/Hd
30	Write Sector(s) (w/ retry)		Valid	Valid	Valid	Drv/Hd
31	Write Sector(s) (w/o retry)		Valid	Valid	Valid	Drv/Hd
32	Write Long (w/ retry)		Valid	Valid	Valid	Drv/Hd
33	Write Long (w/o retry)		Valid	Valid	Valid	Drv/Hd
40	Read Verify Sector(s) (w/ retry)		Valid	Valid	Valid	Drv/Hd
41	Read Verify Sector(s) (w/o retry)		Valid	Valid	Valid	Drv/Hd
50	Format Track		Valid		Valid	Drv/Hd
70-7F	Seek			Valid	Valid	Drv/Hd
90	Execute Drive Diagnostic					Drv*
91	Initialize Drive Parameters		Valid			Drv/Hd
94 E0	Standby Immediate					Drv
95 E1	Idle Immediate					Drv
96 E2	Standby		Valid			Drv
97 E3	Idle		Valid			Drv
98 E5	Check Power Mode		Valid			Drv
99 E6	Set Sleep Mode					Drv
9A	Intel Get/Set Features		Valid	Valid	Valid	Drv/Hd
C4	Read Multiple		Valid	Valid	Valid	Drv/Hd
C5	Write Multiple		Valid	Valid	Valid	Drv/Hd
C6	Set Multiple Mode		Valid			Drv
E4	Read Buffer					Drv
E8	Write Buffer					Drv
EC	Identify Drive					Drv
EF	Set Features	Valid				Drv
80-8F, C0-C3, F0-FF	Vendor Unique	Valid	Valid	Valid	Valid	Drv/Hd

NOTE:
*Addressed to drive 0. Both drives execute the command.

7

Recalibrate, 1X

Recalibrate:

- sets BSY
- selects the flash-array equivalent of cylinder 0
- updates the Status register
- clears BSY
- generates an interrupt.

Read Sector(S), 20 OR 21

This command reads from 1 to 256 sectors as specified in the Sector Count register; a sector count of 0 requests 256 sectors. The Sector Number register specifies the beginning sector.

DRQ is always set prior to data transfer, even in the presence of an error condition.

If an error occurs, read terminates at the erred sector. The sector buffer retains flawed data.

At command completion, command block registers contain the last-read sector's cylinder, head, and sector numbers (or LBA).

Read Long, 22 or 23

The *read long* command is similar to the *read sectors* command but returns the desired sector's data field and ECC bytes (8- bits wide). During a *read long* command, the drive does not check the ECC bytes to determine data errors. Only single- sector read long operations are supported.

Write Sector(s), 30 or 31

This command writes from 1 to 256 sectors as specified in the Sector Count register beginning at the specified sector; a sector count of 0 transfers 256 sectors. The Sector Number register specifies the beginning sector. Buffer data followed by ECC bytes are written to the sector's data field.

If an error occurs during a multiple sector write, writing terminates at the erred sector. The host may read the command block to discern the error and erred sector. See the Error Posting section for Error and Status register contents.

At command completion, command block registers contain the last written sector's cylinder, head, and sector number (or LBA).

Write Long, 32 or 33

This command is similar to the *write sectors* command but writes data and ECC bytes (8-bits wide) directly from the sector buffer; the drive does not generate ECC bytes itself. Only single sector write long operations are supported.

Read Verify Sector(s), 40 or 41

This command, identical to the *read sectors* command, transfers no data but resets DRQ and sets BSY. When the requested sectors are verified, the drive clears BSY and generates an interrupt.

If an error occurs, verify terminates at the erred sector. The Sector Count register contains the number of unverified sectors.

Upon command completion, command block registers contain the last verified sector's cylinder, head, and sector number (or LBA).

Format Track, 50

This command marks a track's sectors as "deleted." The next time a reclamation procedure is invoked, these sectors are erased and returned to the available flash-memory pool.

Cylinder High and Cylinder Low registers specify the track address (or LBA). The Sector Count register specifies the number of sectors. When the command is accepted, the drive sets DRQ and waits for the host to fill the sector buffer. The sector buffer's contents are not written to the media and are, in fact, ignored. The drive then:

- clears DRQ
- sets BSY
- begins command execution.

Each 16-bit buffer word, contiguous from the start of a sector buffer, represents a sector. Buffer words remaining after the last sector representation are filed with zeros. DD_{8-15} contain the sector number. Words appear in the track's sequence. DD_{0-7}'s descriptor value contains:

00h—format sector as good
20h—un-assign this sector's alternate location
40h—assign this sector to an alternate location
80h—format sector as bad.

Seek, 7X

This command selects the flash-array equivalent of the command-specified head and track. The drive need not be formatted for *seek* to work properly. The drive:

- completes the seek
- returns interrupt
- immediately returns successful seek status, DSC = 1.

If the drive receives another command during a *seek*, the drive:

- sets BSY = 1
- waits for seek-complete
- executes the new command.

Execute Drive Diagnostic, 90

This command performs internal self-diagnostic tests; the DRV bit is ignored. The drive:

- posts its diagnostic results
- clears BSY
- generates an interrupt
- loads the Error register status code (Table 18).

Initialize Drive Parameters, 91

This command sets the:

- sectors per track (Sector Count register)
- number of heads (minus 1) per cylinder
- Drive/Head register's drive bit (DRV).

Upon command receipt, the drive

- sets BSY
- saves the parameters
- clears BSY
- generates an interrupt.

This command does not check sector-count and head-value validity. If invalid, no error is posted until another command makes an illegal access.

Standby Immediate, 94 OR E0

This command places the drive in standby mode. The drive sets the interrupt before completing the standby mode transition. If the drive is already in standby, the standby sequence is not re-executed.

Idle Immediate, 95 OR E1

At this command, the drive:

- sets BSY
- enters idle mode
- clears BSY
- generates an interrupt (even if it has not fully transitioned to idle mode).

Standby, 96 OR E2

This command places the drive in standby mode. The drive sets the interrupt before transitioning to standby mode. If the drive is already in standby mode, the standby sequence is not re-executed.

Idle, 97 OR E3

Idle causes the drive to:

- set BSY
- enter idle mode
- clear BSY
- generate an interrupt (even if the drive has not fully transitioned to idle mode).

The spinup sequence is not executed if the drive is already active.

Check Power Mode, 98 OR E5

This command checks the power mode. If the drive is in, going to, or recovering from standby mode, it:

- sets BSY
- clears the Sector Count register
- clears BSY
- generates an interrupt.

If it is in idle mode, the drive:

- sets BSY
- sets the Sector Count register to FFh
- clears BSY
- generates an interrupt.

Set Sleep Mode, 99 OR E6

This command places the drive in sleep mode. The drive:

- powers down all internal circuitry
- clears BSY
- generates an interrupt.

If the drive is already asleep, the sleep sequence IS re-executed.

A drive self-powers-on following a command register write, power-on, hardware reset, or host-issued software reset.

Intel Get/Set Features, 9A

The *Intel get/set features* command, with a sector count register subfunction code, gets or sets special features. A "zero" subfunction code retrieves features, and "one" subfunction code sets features. Table 20 describes the Intel unique features.

Table 20. Intel Vendor Unique Arguments (Command Register = 9A)

Subfunction	Set Features	Sector No.	Sector Count	Cylinder	Drive/Head
Get Features	don't care	9A	0	Intel Vendor Word	don't care
Set Features	don't care	9A	1	Intel Vendor Word	don't care
ATA Code update	Sectors count	9A	5	Security ID = FEED	A1
Low-level Format	don't care	9A	6	don't care	don't care

NOTE:
Cylinder low and high registers = vendor argument low- and high-bytes.

INTEL VENDOR WORD

F	E	D	C	B	A	9	8
Compression	ECC	Auto Sleep	Auto Pwr-dn	Reserved	Power Mode		Master/Slave
Reserved	Reserved	Reset Mode	Reserved	Cache	Command Governing		Drive C
7	6	5	4	3	2	1	0

Bit 0 Drive is Master (Drive C).

Bits 1-2 Max. governing during clean-up.

Bit 2	Bit 1	Reclamation Time (seconds)
0	0	2
0	1	4

Bit 2	Bit 1	Reclamation Time (seconds)
1	0	4
1	1	8

Bit 3 Enable Caching.

Bit 4, 6-7, B Reserved.

Bit 5 Enable special reset mode (1).

Bit 8 Master/slave; Non-Conner = 0, Conner = 1.

Bits 9-A Reclamation state machine (RSM) power modes.

Bit A	Bit 9	Total RSMs	Read RSMs	Write RSMs	Perf. Rating	Power Rating
0	0	1	0	0	lowest	lowest
0	1	1	1	0		
1	0	2	1	1		
1	1	2	2	1	highest	highest

Bit C Auto power-down; 0 = enabled; 1 = disabled.

Bit D Auto standby = 0, auto sleep = 1 (default).

Bit E ECC enabled (0) or disabled (1).

Bit F compression; enabled = 0, disabled = 1 (not supported).

Read Multiple, C4

The *read multiple* command is similar to the *read sector* command. An interrupt is not generated by every sector but by a block transfer. The *set multiple mode* command, which must be executed prior to the *read multiple* command, determines the sector block count to be transferred without intervening interrupts. DRQ qualification is required only at the data block's start, not at each sector.

The Sector Count register contains the number of sectors requested (not the block count). If a non-even block count is requested, all possible full blocks are transferred followed by a partial block; remainder of sector count ÷ block count.

The *read multiple* command is rejected with an aborted-command error if it is executed before a *set multiple mode* command or if *read multiple* commands are disabled.

During *read multiple* commands, encountered disk errors are posted at the beginning of a block or partial block transfer. DRQ is still set and the data, including corrupted-data, transfers normally.

Command block register contents are undefined following a sector-in-error data block transfer. The host can obtain error information by retrying the transfer as individual requests.

Subsequent or partial blocks are transferred only if the data-error was correctable. Other errors stop the command after transferring the error-containing block. Interrupts are generated when the beginning of each block or partial block transfer sets DRQ.

Write Multiple, C5

This command is similar to the *write sectors* command. The drive sets BSY within 400 nsec after the command. Interrupts don't occur after each sector transfer but after a block transfer. The *set multiple* command, which must be executed prior to the *write multiple* command, defines the number of sectors transferred without intervening interrupts. DRQ transfer-qualification is required only at the data block start, not at each sector.

The Sector Count register contains the number of sectors requested (not the block count). If a non-even block count is requested, all possible full blocks are transferred followed by a partial block; remainder of sector count ÷ block count.

The *write multiple* command is rejected with an aborted-command error if it is executed before a *set multiple mode* command or if *write multiple* commands are disabled.

Disk errors during *write multiple* commands are posted after the attempted block or partial block disk write. The *write multiple* command ends with the erred sector, even if it was in the middle of a block; subsequent blocks are not transferred. Setting DRQ generates interrupts at the beginning of each block or partial block.

Command block register contents are undefined following a sector-in-error data block transfer. The host should retry the transfer as individual requests to obtain error information.

Set Multiple Mode, C6

This command lets the drive perform read- and write-multiple operations and establishes the command block count. At the command, the drive sets BSY and checks the Sector Count register.

The Sector Count register receives the sectors-per-block number. The flash drive supports 2, 4, 8, and 16 sector block sizes.

If the Sector Count register value is valid and the block count is supported, command execution is enabled and subsequent *read-* and *write-multiple* commands use this value. If a block count is not supported or the Sector Count register contains 0, an aborted-command error is posted and *read-* and *write-multiple* commands are disabled.

At power-on, or after hardware reset, *read-* and *write-multiple* modes are disabled.

Read DMA, C8 OR C9

This command is not supported.

Write DMA, CA OR CB

This command is not supported.

Read Buffer, E4

This command allows the host to read sector buffer contents. When issued, the drive:

- sets BSY
- sets the sector buffer for a read operation
- sets DRQ
- clears BSY
- generates an interrupt.

The host then reads up to 512 data bytes. The drive synchronizes write and read buffer commands so sequential commands access the same 512 buffer bytes.

Write Buffer, E8

This command allows the host to overwrite the drive's sector buffer data pattern. The drive synchronizes sequential *write* and *read buffer* commands so they access the same buffer bytes.

Identify Drive, EC

The *identify drive* command presents drive parameter information to the host. When issued, the drive:

- sets BSY
- transfers parameter information to the sector buffer
- sets DRQ
- generates an interrupt.

The host then reads the sector buffer. Table 21 defines arguments and their meanings. Reserved bits or words are zero.

Identify Drive Arguments

Table 21. Identify Drive Arguments

Word(1)	Bit	Value (5M/10M)	Description
0	15	848C	1 = non-magnetic drive
	14	(General	0 = format speed tolerance gap not required
	13	configuration	0 = track offset option available
	12	bit-significant	0 = data strobe offset option available
	11	information)	0 = rotational speed tolerance is > 0.5%
	10		1 = disk transfer rate > 10 MB/s
	9		0 = disk transfer rate > 5 MB/s but ≤ 10 MB/s
	8		0 = disk transfer rate ≤ 5 MB/s
	7		1 = removable cartridge drive
	6		0 = fixed drive
	5		0 = spindle motor control option implemented
	4		0 = head switch time > 15 μs
	3		1 = not MFM encoded
	2		1 = soft sectored
	1		0 = hard sectored
	0		0 = reserved
1(2)		009C/0140	Number of cylinders in the default translation mode
3(2)		0002	Number of heads in the default translation mode
4(2)		4000	Number of unformatted bytes per track in the default translation mode
5(2)		0200	Number of unformatted bytes per sector in the default translation mode
6(2)		0020	Number of sectors per track in the default translation mode
10–19		Unique	Serial No. 20 ASCII chars, right justified, space padded (20h); 0000h = none
20		0003	Buffer Type: Dual port multi-sector; simultaneous transfers with read caching
21		0040	Buffer size in 512-byte increments (0000h = not specified)
22		0004	Number of ECC bytes passed on *read/write long* commands (0000h = none)
23–26		Unique	Firmware rev.; 8 ASCII chars, left justified, space padded (20h); 0000h = none
27–46(3)			Model No. 40 ASCII chars, left justified, space padded (20h); 0000h = none
47	7–0	8008	Max *read/write multiple* sectors transferred/interrupt (00h = not allowed)
48		0000	0000h = cannot perform doubleword I/O; 0001h = can perform doubleword I/O
49	15–10	0201	Capabilities; 0 = reserved
	9		1 = LBA mode supported; 0 = CHS mode supported
	8		1 = DMA supported; 0 = DMA not supported
51(4)	15–8	0100	PIO data transfer cycle timing mode
52	15–8	0100	DMA data transfer cycle timing mode (not supported)
54		009C/0140	Number of current cylinders in the current translation mode
55		0002	Number of current heads in the current translation mode
56		0020	Number of current sectors-per-track in the current translation mode
57–58		2700/5000	Sector capacity, excluding device-specific sectors, = words (54 * 55 * 56)
59	8	0000	1 = Multiple sector setting, bits 7–0 are valid
	7–0		If bit 8 = 0; Sectors transfered per interrupt on *read/write multiple* commands
60–61		26FF/4FFF	Total user addressable sectors if LBA mode is supported, CHS mode = 0
62	15–8	0203	Single-word DMA transfer mode active, indicated by bits 7–0; (not supported)
	7–0		Single-word DMA transfer modes available; (not supported)
63	15–8	0203	Multi-word DMA transfer mode active, indicated by bits 7–0; (not supported)
	7–0		Multi-word DMA transfer modes available; (not supported)

NOTES:
1. Words not listed are ATA or Intel reserved.
2. Initialize drive parameters does not affect words 1–6.
3. 5MB drive = iFD005P2SA, 10MB drive = iFD010P2SA.
4. Bits 15–8 determine parametric timing specification categories, The flash drive uses mode 1 which transfers 5.22 MB/s.

Set Features, EF

The host sets features by writing values to the Set Feature register and EFh to the command register.

55 Disable read look-ahead

81 Disable 8-bit data transfers (not supported)

82 Disable write cache (not supported)

AA Enable read look-ahead (not supported)

CC Enable power-on defaults

Error Posting

Table 22 defines each command's valid errors. See Error and Status register definitions.

Table 22. Register Contents During Error Posting

Command	Error Register						Status Register				
	BBK	UNC	IDNF	ABRT	T0NF	AMNF	DRDY	DWF	DSC	CORR	ERR
Check Power Mode				✓			✓	✓	✓		✓
Execute Drive Diags											✓
Format Track			✓	✓			✓	✓	✓		✓
Identify Drive				✓			✓	✓	✓		✓
Idle				✓			✓	✓	✓		✓
Idle Immediate				✓			✓	✓	✓		✓
Initialize Drive params							✓	✓	✓		
Recalibrate				✓	✓		✓	✓	✓		✓
Read Buffer				✓			✓	✓	✓		✓
Read Long	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓
Read Multiple	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓
Read Sector(s)	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓
Read Verify Sector(s)	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓
Seek			✓	✓			✓	✓	✓		✓
Set Features				✓			✓	✓	✓		✓
Set Multiple Mode				✓			✓	✓	✓		✓
Sleep				✓			✓	✓	✓		✓
Standby				✓			✓	✓	✓		✓
Standby Immediate				✓			✓	✓	✓		✓
Write Buffer				✓			✓	✓	✓		✓
Write Long	✓		✓	✓			✓	✓	✓		✓
Write Multiple	✓		✓	✓			✓	✓	✓		✓
Write Sector(s)	✓		✓	✓			✓	✓	✓		✓
Invalid Command				✓			✓	✓	✓		✓

NOTE:
✓ = Valid for this command

PROTOCOL OVERVIEW

Command classes are grouped according to protocols described for command execution. For all commands, the host must first check for BSY = 0 before proceeding further. For most commands, the host should not proceed until DRDY = 1. The host can execute commands shown with DRDY = x when DRDY = 0. Data transfers may occur in ways not described below, but should work with all known ATA drive implementations.

PIO Data In Commands

This class includes:

- *Identify drive*
- *Read buffer*
- *Read long*
- *Read sector(s)*

Execution includes one or more 512 byte data-sector drive-to-host transfers (512 + ECC bytes if *read long*). If the drive presents error status, it prepares to transfer data at the host's discretion.

a) The host writes parameters to the Features, Sector Count, Sector Number, Cylinder, and Drive/Head registers

b) The host writes the Command register's command code

c) The drive sets BSY and prepares for data transfer

d) When a data sector is available, the drive sets DRQ, clears BSY, and asserts interrupt

e) At interrupt, the host reads the Status register, the drive negates interrupt, and the host reads one data sector from the Data register

f) The drive clears DRQ

g) If another sector transfer is required, the drive sets BSY and repeats the sequence from d).

PIO Read Command

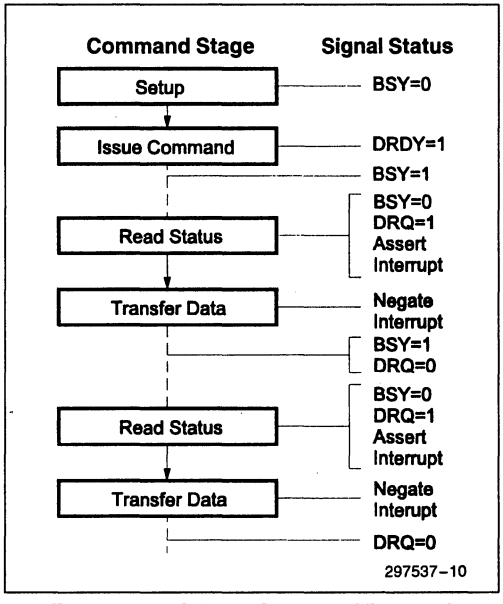


Figure 10. PIO Read Command Protocol

PIO Read Aborted Command

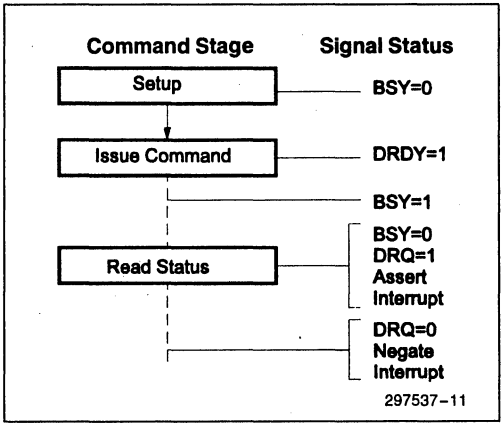


Figure 11. PIO Read Abort Command Protocol

Although DRQ = 1, no data is transferred.

7

PIO Data Out Commands

This class includes:

- *Format*
- *Write buffer*
- *Write long*
- *Write sector(s)*

Execution includes one or more 512 byte host-to-drive data-sector transfers (512 + ECC bytes if *write long*).

a) The host writes parameters to the Features, Sector Count, Sector Number, Cylinder, and Drive/Head registers

b) The host writes the Command register's command code

c) The drive sets DRQ when it can accept the first sector of data

d) The host writes one sector of data to the Data register

e) The drive clears DRQ and sets BSY

f) At sector processing complete, the drive clears BSY and asserts interrupt. If another sector transfer is required, the drive also sets DRQ

g) The host reads the Status register after detecting interrupt

h) The drive negates the interrupt

i) If another sector transfer is required, the sequence repeats from d).

PIO Write Command

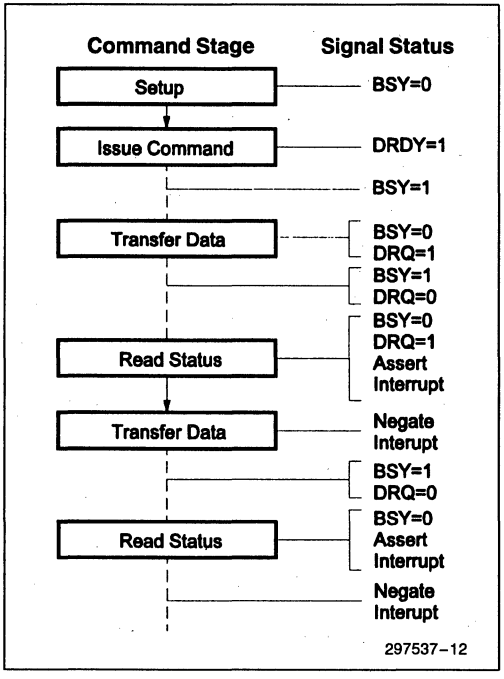


Figure 12. PIO Write Command Protocol

PIO Write Aborted Command

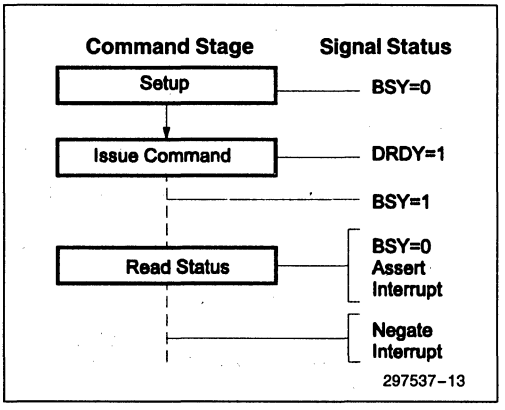


Figure 13. PIO Write Abort Command Protocol

Non-Data Commands

This class includes:

- *Execute drive diagnostic* (DRDY = x)
- *Idle*
- *Initialize drive parameters* (DRDY = x)
- *Read power mode*
- *Read verify sector(s)*
- *Recalibrate*
- *Seek*
- *Set features*
- *Set multiple mode*
- *Standby*

Command execution involves no data transfer.

a) The host writes parameters to the Features, Sector Count, Sector Number, Cylinder, and Drive/Head registers
b) The host writes the Command register's command code
c) The drive sets BSY
d) When the drive completes sector processing, it clears BSY and asserts interrupt
e) The host reads the Status register after detecting interrupt
f) The drive negates the interrupt.

Miscellaneous Commands

This class includes:

- *Read multiple*
- *Sleep*
- *Write multiple*
- *Write same*

See command descriptions for individual protocols.

DOS Data Format

The ATA interface provides two primary benefits: 1) the media is hidden from the host so it doesn't have to deal with magnetic- or flash-based eccentricities and 2) any filing system that comprehends sector-size storage elements can be used. Although the ATA standard was developed for Intel-architecture and DOS-based machines, it does not specify or recommend a file format. An ATA drive can work with any hardware architecture or software operating system.

The PCMCIA-ATA specification helps reinforce this platform independence. PCMCIA-ATA compliant drives aren't restricted to the PC-AT's traditional primary and secondary addressing, but can operate at any I/O or memory address. To allow for plug-and-play interchangability, system designers should implement a filing system that is consistent among planned environments.

For example, a simple embedded system at a remote location may log data to a flash drive. The drive can be removed and its data downloaded to a laptop or desktop computer. The data-logging system must store data in a format that's consistent with the download system. Since DOS provides the most widely used filing system, many embedded and non-DOS-architecture systems use a DOS-compatible file format for greater plug-and-play interchangability.

Disk Layout

A DOS-formatted disk is divided into several sections:

- partition table (hidden from the file system)
- boot sector(s)
- file Allocation Table 1 (FAT)
- file Allocation Table 2 (FAT)
- root directory
- data.

Figure 14 shows an example 10MB flash drive's layout

C	H (hex)	S	Physical Sector Drive X	LBA (hex)
00	0	1	Partition Table (20h = 32d hidden sectors)	0001
00	1	1	Boot Sector (1 sector)	0021
00	1	2	FAT 1 (8 sectors)	0022
00	1	A	FAT 2 (8 sectors)	002A
00	1	12	Root Directory (20h = 32d sectors)	0032
01	0	12	Data (4FAFh = 20399d sectors)	0052
60	1	3F		5000

Figure 14. Example 10MB flash drive's layout.

Disk Partitioning

DOS provides an FDISK utility that uses BIOS functions to partition an IDE-ATA drive, establishes the file system, and makes partitions bootable. If the flash drive is used in its IDE-ATA mode, FDISK can partition it. If it's used in a PCMCIA socket, which uses device driver software rather than BIOS, an FDISK-type utility such as SystemSoft's ATAINIT, Phoenix' PCMFDISK, Award's PCPREP, or VMI's PCFORMAT can partition it. Because of the relatively small size of today's PCMCIA cards, most PCMCIA partitioning utilities only create one partition. This will probably change as future larger drives become available and as various operating systems adopt the PCMCIA-ATA standard.

To ensure that a flash drive can be used in the widest variety of systems, it may be best to partition it for the earliest applicable DOS version, i.e. DOS 3. This limits partition size to 32MB. To accomplish the same goal with later DOS versions, the first partition can be limited to 32MB and serve as the main interchange partition.

The main, and up to three extended-partition tables, occupy a drive's (and each partition's) first sector, which is otherwise hidden from the file system. It provides:

- partition boot/non-boot status
- starting and ending cylinder/head/sector
- partition type (e.g. 12- or 16-bit FAT)
- partition size.

Table 23. Example 10MB ATA Drive Partition Table

Partition Table (C, H, S, = 0, 0, 1)	Offset	Main (required)	Offset	Extended 1 (optional)	Offset	Extended 2 (optional)	Offset	Extended 3 (optional)
Boot, non-boot(1)	1BE	80	1CE	00	1DE	00	1EE	00
Starting head	1BF	01	1CF	00	1DF	00	1EF	00
Starting cyl, sector(2)	1C1, 1C0	00, 01	1D1, 1D0	00, 00	1E1, 1E0	00, 00	1F1, 1F0	00, 00
Partition type(3)	1C2	01	1D2	00	1E2	00	1F2	00
Ending head	1C3	01	1D3	00	1E3	00	1F3	00
Ending cyl, sector(2)	1C5, 1C4	3F, 60	1D5, 1D4	00, 00	1E5, 1E4	00, 00	1F5, 1F4	00, 00
First logical sector	1C9-1C6	0000 0020	1D9-1D6	0000 0000	1E9-1E6	0000 0000	1F9-1F6	0000 0000
Partition length (sec)	1CD-1CA	0000 4FE0	1DD-1DA	0000 0000	1ED-1EA	0000 0000	1FD-1FA	0000 0000
Signature bytes							1FF, 1FE	AA, 55

NOTES:
1. Bootable disk = 80, non-bootable = 00.
2. Sector number is 6 bits, cylinder number is 10 bits (most significant two bits are bits 6 and 7 of bytes 1x0 or 1x4)
3. Partition type; 01 = primary DOS 12-bit FAT, 04 = primary DOS 16-bit FAT, 05 = extended DOS, 08 = other non-DOS

Boot Sector

The boot sector (Table 24) follows the partition table. In addition to boot-time text messages, it describes the partition's sector, track, root directory structure, and FAT tables.

Table 24. Example 10MB ATA drive Boot Sector

Boot Sector (C, H, S = 0, 1, 1 from partition table)	Offset	Logical Drive 0
Bytes per sector	0C, 0B	0200
Sectors per cluster	0D	08
Boot sector size	0F, 0E	0001
Number of FATs	10	02
Number of root directory entries	12, 11	0200
Total sectors (if drive size <32MB)	14, 13	4FE0
Media descriptor (F8 = fixed disk)	15	F8
Sectors per FAT	17, 16	0008
Sectors per track	19, 18	0020
Number of heads	1B, 1A	0002
Number of hidden sectors	1D, 1C	0020
Total sectors (if drive size >32MB)	23–20	0000 0000

7

FAT Tables

DOS maintains two copies of the File Allocation Table (FAT). The FAT is a linked list that keeps track of clusters used by files. Each FAT entry corresponds to a cluster of sectors in the data-area. A cluster may be one or more sectors.

Tables 23 and 24, for the 10MB drive example, indicate that:

- each FAT entry or cluster pointer is 12-bits (1½ bytes)
- each FAT has 8 sectors (4096 bytes)
- each cluster contains 8 sectors.

The FAT table can point to 4096 ÷ 1½ = 2730 clusters or 21,840 sectors (2730 clusters * 8 sec/cluster) or 11,182,080 bytes. Since the drive actually has only 20,399 sec * 512 bytes/sec = 10,444,288 bytes, not all cluster pointers are used.

Root Directory

The boot sector describes the number of root directory entries. Table 24's example shows 0200h = 512 entries. Each DOS root directory entry contains 32 bytes. So, 512 32-byte entries makes the root directory size 32d (20h) sectors. Figure 15 shows byte assignments.

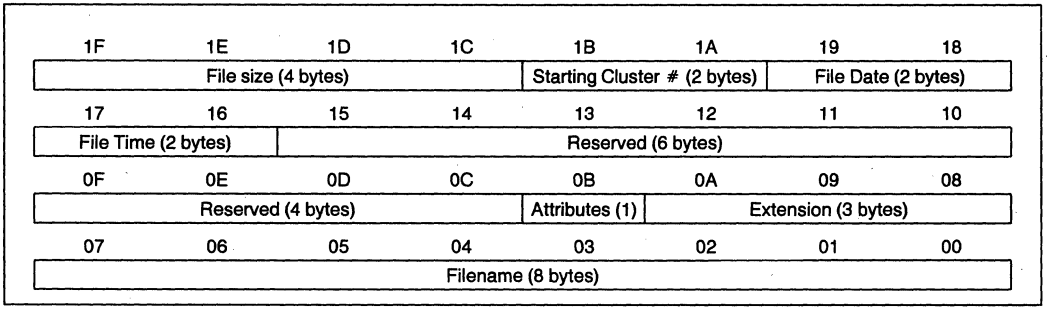


Figure 15. A DOS 32-byte Root Directory Entry

Bytes 0–7 ASCII text file name

Bytes 8–A ASCII text file extension

Byte B Indicates file attribute if bit is set (1)

7	6	5	4	3	2	1	0
Unused	Unused	Archive	Subdirectory	Volume Label	System	Hidden	Read-only

Bytes C–15 Reserved, should be zeros

Bytes 16–17 Coded file creation/modification time = (hour x 2048) + (minutes x 32) + (seconds ÷ 2)

Bytes 18–19 Coded file creation/modification date = [(year-1980) x 512] + (month x 32) + day

Bytes 1A–1B Points to file's starting cluster

Bytes 1C–1F File size; number of bytes. This size may differ from the actual size required to store the file since files are stored in cluster-size chunks. In the 10MB drive example, a 245-byte file would actually consume 1 cluster or 4096 bytes.

Disk Formatting

A root directory entry's Starting Cluster Number points to a file's beginning cluster. The file's starting sector can also be derived from this number. The Starting Cluster Number corresponds to an entry in the FAT table which points to the file's next cluster, which points to the next cluster, etc. Figure 16 shows an example 12-bit FAT linked list; it assumes that the first root directory entry's starting cluster pointer value is 0002 (it points to cluster number 002).

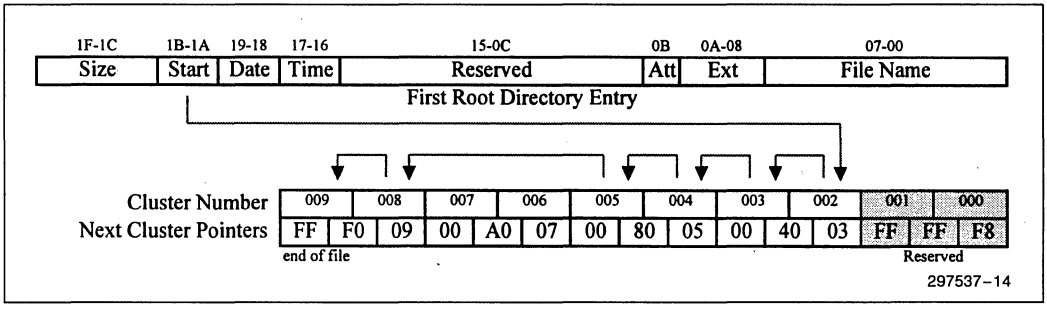


Figure 16. The Root Directory Entry Points to a FAT cluster entry which begins the files linked-cluster-list.

A 12-bit FAT entry has three-nibbles (16-bit fat entries, of course, have four-nibbles). Note that the FAT's first two cluster pointers (000 and 001) contain value FFFFF8. The first byte, F8, describes the media and should be the same as the partition table's media descriptor (byte 15h). The second and third bytes are always FFFF. In the example, the root directory entry points to the file's first linked-list cluster, which is 002. Cluster 002's value points to cluster 3. Cluster 003's value points to cluster 4, and so on. Cluster 005's value points to cluster 8; this indicates that clusters 006 and 007 contain some other file's data and that both files are fragmented. Cluster 008's value points to cluster 9 who's FFF value indicates the file's last cluster. A 000 cluster pointer indicates that the cluster is not assigned and can receive new data. A 16-bit FAT's structure is the same but cluster-pointers are two bytes rather than three nibbles.

LBA Mode

Logical Block Addressing allows a host to deal with a drive as if it is a contiguous string of sectors. Since most disk drives internally operate in LBA mode, it's a logical extension to allow host BIOSs and drivers to use LBA addressing. Neither the host nor drive needs to convert a logical block address into a cylinder, head, sector address. LBA addresses, unlike CHS, start with sector zero (0). The last sector is the partition length (from partition table bytes 1xD-1xC) or the total sectors (from boot sector bytes 14h–13h or 23h–20h) minus 1.

CHS Mode

CHS is the traditional way to access logical drive sectors. The host must convert the LBA address into a CHS value before accessing the drive. Similarly, the drive must convert the CHS to an LBA address. To convert LBA to a CHS value, use the following equation (where sectors-per-cylinder = number-of-heads x sectors-per-track. Number-of-heads and sectors-per-track are obtained from boot sector bytes 1Bh-18h):

Cylinder = integer[(LBA sector + 1) ÷ sectors-per-cylinder]

Head = integer{remainder[(LBA sector + 1) ÷ sectors-per-cylinder] ÷ sectors-per-track}

Sector = remainder{remainder[(LBA sector + 1) ÷ sectors-per-cylinder] ÷ sectors-per-track}

To convert CHS to LBA values, use the following equation:

LBA = [(*Cylinder* x number-of-heads + *Head*) x sectors-per-track] + *Sector*-1

LOW POWER MODES

Intel's flash drive is compatible with standard PCMCIA and ATA low-power modes. These modes are implemented for compatibility but have improved characteristics over standard hard disk drive implementations. These low-power mode improvements make the flash drive ideal for battery-based applications.

Idle

The *Idle* command initiates idle mode. If requested by the host, the drive places itself in idle mode for about 1.5 seconds. I_{CC} maximum current applies only when a host *idle* command forces the drive into idle mode. If no activity is detected, it transitions to sleep mode (or self-initiated deep power-down mode). The drive wakes upon receiving any command or soft/hardware reset.

When it receives an *idle* command, the drive sets BSY, enters idle mode, clears BSY, and generates an interrupt (even if the drive hasn't fully transitioned to idle mode). The power-up sequence is not executed if the drive is already active.

The host can check for idle mode by issuing a *check power mode* command. If it is in idle mode, the drive sets BSY, sets the Sector Count register to FFh, clears BSY, and generates an interrupt.

Standby

The *standby* command initiates standby mode. If requested by the host, the drive places itself in standby mode for about 1.5 seconds. I_{CC} maximum current applies only when a host *standby* command forces the drive into standby mode. If no activity is detected, it transitions to sleep mode (or self-initiated deep power-down mode). The drive wakes upon receiving any command or soft/hardware reset.

When it receives a *standby* command, the drive sets the interrupt before completing the standby mode transition. If the drive is already in standby, the standby sequence is not re-executed.

The host can check for standby mode by issuing a *check power mode* command. If the drive is in, going to, or recovering from standby mode, it sets BSY, clears the Sector Count register, clears BSY, and generates an interrupt.

Sleep

Host-initiated Sleep mode places all flash drive circuitry in a power-down state when the host executes a *set sleep mode* command. A hardware or software reset brings the drive out of sleep mode.

Self-initiated sleep (or deep power-down) mode places non-critical circuitry, including the flash array and programming power unit, into a power-down mode (which is transparent to the host) when the drive detects no activity for about 1.5 seconds. The drive powers up and responds to a host command within 10 ms. The drive also wakes upon receiving a soft or hardware reset.

PCMCIA Power-Down

The flash drive can enter sleep mode if the host sets the Configuration and Status Register's PwrDn bit (bit 2). The drive finishes its in-progress operation. The host must check Rdy/Bsy# to ensure that the drive is ready and is transitioning to sleep mode.

SYSTEM CONSIDERATIONS

Card Type

PCMCIA defines three card types, or thickness. All have the same length and width dimensions—85.6mm x 54.0mm (3.370" x 2.126"). Intel's Flash Drive uses type II, 5.0mm (.196") thickness, card case.

Power Requirements

Intel's Flash Drive requires only 5V ± 5% on its V_{CC} inputs. The drive internally converts V_{CC} voltage to 12V for program and erase operations. The drive works in systems that supply 12V on the V_{PP} pins. Since these pins are not internally connected, the flash drive ignores V_{PP} voltage and will not consume power from V_{PP} pins.

Bootable PCMCIA Drive

Most DOS-machine PCMCIA implementations use drivers that are loaded by CONFIG.SYS. They boot from IDE-ATA drives and don't have knowledge about their PCMCIA sockets until after the boot sequence. Thus, it's impossible for these systems to boot from drives in their PCMCIA sockets.

However, a system with a PCMCIA-aware BIOS can configure its PCMCIA socket(s) before boot-up. Once the BIOS configures the socket controller to decode the Primary-Drive address (1F0–1F7), determines that a flash drive is present, and configures the flash drive into its Primary-Drive address mode, the system can boot from a PCMCIA-socketed flash drive. For a more general solution, the BIOS can contain Socket- and Card-Services. This allows a PCMCIA socket to contain a bootable drive and accommodate other PCMCIA cards.

Intel's Flash Drive can also be used as an IDE-ATA bootable drive. See the IDE-ATA Interface section for details.

Card Insertion and Removing

PCMCIA cards' main benefit is plug-and-play (PnP) removability. Intel's Flash Drive can be removed or inserted into a version 2 PCMCIA-compliant socket while a system is operating.

PCMCIA socket firmware and hardware should monitor the socket's card detect pins (CD1# and CD2#) to determine card presence. If no card is present, socket power (V_{CC} and V_{PP}) should be turned off and address, data, and control signals should be high-Z. This allows card insertion into a "cold" socket without fear of damaging either socket or card hardware.

Once the system detects card insertion or presence, it can apply V_{CC} (and V_{PP} if required) and enable address, data, and control signal buffers. The system should then reset the card, wait an appropriate amount of time for initialization, read the card's CIS, and configure it before accessing ATA registers.

"Warm" socketing (power is off but the socket's address, data, and control signals are active) is discouraged. Although "warm" PCMCIA sockets are much cheaper to implement, the potential for system and card damage is high. No matter how many warnings are included in a system's documentation or on the device itself, users will always forget and remove or insert a PCMCIA card in warm sockets. Even if physical damage doesn't occur, signal disturbance almost certainly will cause system malfunction.

"Hot" socketing (power is applied and the socket's address, data, and control signals are active) is absolutely discouraged. This poor design practice is sometimes attempted in very low-cost designs. A "hot" socket design should be implemented only if the user cannot readily remove the card; e.g. it's embedded inside the system.

Dealing with Existing System Drives

In PCMCIA implementations, the flash drive's flexible memory and I/O addressing modes allow it to work without conflicting with memory, I/O, or interrupt resources. In DOS systems, a bootable drive will occupy the primary drive address (1F0h–1F7h, 3F6h, 3F7h) and IRQ_{14}. Other system drives may occupy the primary drive address (as a slave) and/or secondary drive address (170h–177h, 376h, 377h, master and slave) and IRQ_{14}. When system software allocates resources for a PCMCIA-socketed flash drive, it should be configured for a memory or I/O address and IRQ that avoid the system drive(s).

Some host PCMCIA socket adapter chips cause a conflict between floppy-disks and PCMCIA-ATA drives when implementing addresses 3F7h or 377h. Floppy drives use bit 7 at these addresses as a Media Change status bit. ATA drives use these addresses for the Drive Address register; bit 7 is high-Z. However, since PCMCIA socket adapter chips are general purpose, they may not float bit 7 at I/O addresses 3F7h or 377h. Since the ATA-drive's Drive Address register's information is rarely useful and can be determined from other ATA registers, it is recommended that it not be allocated or used by PCMCIA-ATA drivers.

In IDE-ATA implementations, the flash drive must be either the primary master drive, at primary drive address (1F0h–1F7h, 3F6h, 3F7h) and IRQ_{14}, or a master drive at some other system I/O location, e.g. secondary drive address (170h–177h, 376h, 377h). Present flash drives do not implemented Slave-mode.

PCMCIA Drivers

PCMCIA-ATA drive implementation requires three system software layers:

- Socket Services
- Card Services (with resource allocation utility)
- ATA driver

BIOS vendors provide all three layers in their PCMCIA offerings. Intel also provides drivers and their reference code that support all three layer implementations. Figure 17 summarizes the PCMCIA software stack.

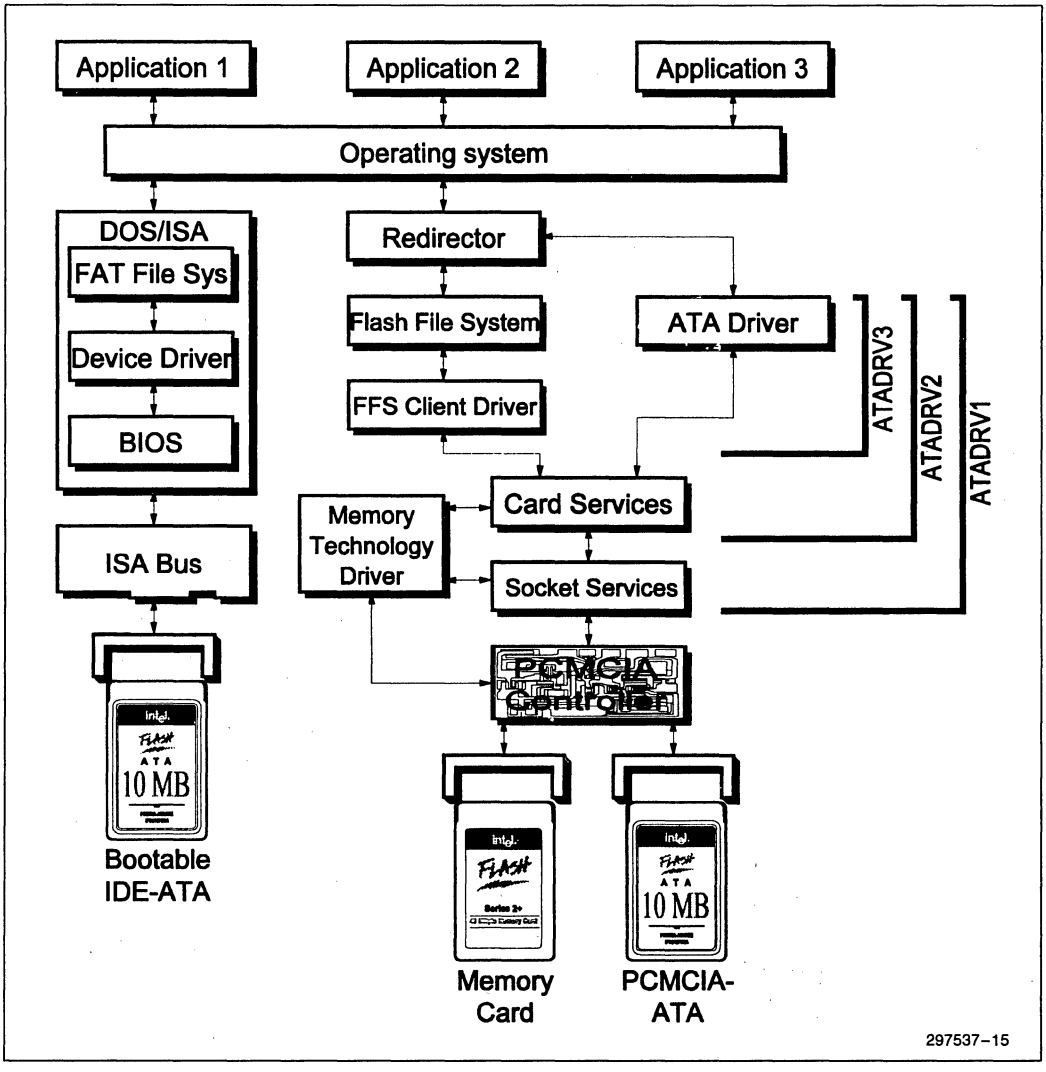


Figure 17. PCMCIA Software Stack

Socket Services

Since PCMCIA does not specify socket implementation, hardware vendors are free to implement adapters in unique and creative ways. Usually, socket adapter manufacturers provide socket services that support their hardware. PCMCIA defined a Socket Services protocol that isolates PCMCIA-adapter hardware from host software. This allows host software to access hardware through standard function calls without knowing how the hardware actually implements the socket interface. Socket Service's limited functionality can turn socket power on and off and map (or address-decode) PC card memory, I/O, and interrupt resources. A hierarchical layer, called Card Services, uses Socket Service's standardized software interface to determine and configure a socket's capabilities.

Card Services

Card Services allocates, controls, and arbitrates system memory, I/O, and interrupt resources requested by "client" drivers (or application programs). As the interface layer between Socket Services and the client software, it makes clients simpler to write and implement. Card Services notifies clients of card insertion/removal and power management requests. Since it allocates system resources, it also prevents conflicts between clients that try to use the same resources.

Resource Allocation Utilities

Several ISV BIOS vendors implement resource allocation utilities that work with Card Services to determine available system memory, I/O, and interrupt resources. These utilities usually build text files during PCMCIA-suite installation that specify available and reserved resources. The four major ISV initialization files are:

- American Megatrends Inc. (AMI)—AMICS.CFG
- Award—CARDWARE.INI
- Phoenix—PCM.INI
- SystemSoft—CSALLOC.INI

Older versions of these suites may use CONFIG.SYS command-line switches to assign system resources.

ATA Driver

Ideally, a system could use its BIOS-ATA driver to communicate with a PCMCIA-ATA drive. However, since DOS-system CONFIG.SYS files usually setup PCMCIA sockets and install device drivers after the BIOS establishes resident disk drive communication, it's very difficult to redirect BIOS routines to PCMCIA devices. Therefore, systems use PCMCIA-ATA device drivers that mirror the functionality of BIOS-ATA drivers to communicate with socketed drives. The four major ISV PCMCIA-ATA client driver files are:

- American Megatrends Inc. (AMI)—PCATA.SYS
- Award—PCDISK.EXE
- Phoenix—PCMATA.SYS
- SystemSoft—ATADRV.EXE

Intel PCMCIA- and IDE-ATA Drivers

Intel provides PCMCIA-ATA drivers that support all PCMCIA-compliant ATA drives; rotating hard disk and flash drives. These drivers allow system developers to quickly evaluate Intel's Flash Drive and can be licensed for inclusion in end products. They come with fully documented source code and software-designers' guides. Three drivers have been developed that serve different system configurations. Contact Intel for drivers, source code, and software reference design guides.

ATADRV1

ATADRV1 is a compact, DOS-based, high-performance PCMCIA-ATA driver. It sets up an i82365SL-compatible socket controller and communicates directly with an ATA drive. It does not require Socket- or Card-Services. Since it is a stand- alone PCMCIA-ATA driver, it consumes minimal RAM footprint. System manufacturers that create systems with i82365SL-compatible socket controllers can quickly support Intel's PCMCIA-ATA flash drive without purchasing Socket- or Card-Services from third-party sources. System developers can modify source code for custom implementations or to support other socket controller chips. The driver's limitations are: it is written for DOS-based, Intel-architecture machines, it requires an i82365SL-compatible socket controller, and, since it does not utilize Socket- or Card-Services, may not work with other PCMCIA cards.

ATADRV2

ATADRV2 is similar to ATADRV1 but communicates with a PCMCIA-ATA drive and requests socket controller resources through Socket Services. It does not require Card-Services. Its RAM footprint is similar to ATADRV1, but Socket Services plus ATADRV2 may consume more total resources. System manufacturers that create systems with other than i82365SL- compatible socket controllers can quickly support Intel's PCMCIA-ATA flash drive without purchasing Card-Services from third-party sources. System developers can modify source code to create custom implementations and include Socket Services provided by the controller manufacturer. The driver's limitations: it is written for DOS-based, Intel-architecture machines and, since it does not utilize Card-Services, may not work with other PCMCIA cards.

ATADRV3

ATADRV3 is similar to ATADRV2 but communicates with a PCMCIA-ATA drive and requests socket controller resources through Card Services. Its RAM footprint is similar to ATADRV1and ATADRV2, but Socket Services plus Card Services plus ATADRV3 may consume more total resources. System manufacturers that create systems with any socket controller can quickly support Intel's PCMCIA-ATA flash drive if Socket- and Card-Services are purchased from third-party sources. System developers can modify source code to create custom implementations. Since ATADRV3 uses Socket- and Card- Services, other PCMCIA cards can be used as well as any PCMCIA-ATA drive. ATADRV3 can also replace older PCMCIA-ATA drivers that only support one vendor's cards.

CONCLUSION

This design guide covers many flash, ATA, PCMCIA, and system related issues. Most flash drive implementors don't need to worry about most of them. With ISV and Intel BIOS- and PCMCIA-ATA drivers so readily available, it's easy to implement ATA solutions that support flash drives.

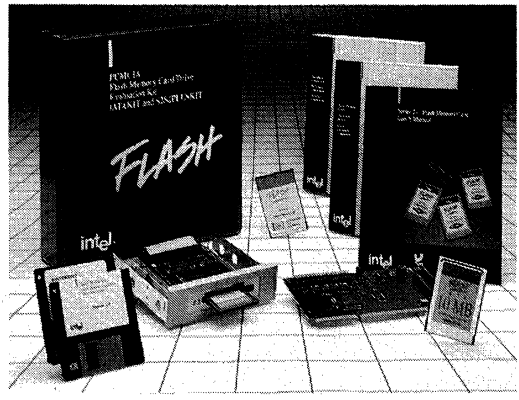
Flash drives provide rugged, high-density, high-performance, low-power, mass-storage solutions that hide the flash media behind the industry-standard ATA interface. Host systems don't even realize or need to be aware that they're storing/retrieving data to/from flash memory.

For embedded, non-DOS, or non-Intel-architecture systems, this design guide contains valuable information about how systems should interact with ATA cards and how flash drive cards implement PCMCIA and ATA registers and commands. This guide, coupled with the ATA and PCMCIA specifications, should provide all the information needed to support flash drives in today's portable systems.

Flash Memory Card and Flash Drive Evaluation Kit Product Brief

Product Highlights

- Complete kit for adding a PCMCIA socket to an AT-based system
- The Flash Drive Kit (iATAKIT) includes a 5MB Flash Drive
- The Flash Card Kit (S2S2PLUSKIT) includes both a 4MB Series 2+ and a 2MB Series 2 Flash Card
- Includes ISA-based PCMCIA Reader/Writer that fits into an empty floppy disk slot on your system
- Created for x86 DOS- or Windows*-based systems
- Includes source code for non-DOS system development
- Includes SystemSoft's* CardSoft* suite of software
- Contains industry-standard PCMCIA software, including SystemSoft Flash File System (FFS) software
- Contains a 68-pin PCMCIA-ATA to 40/44-pin IDE adapter for easy conversion (iATAKIT only)
- Excellent for evaluating Flash Card or Flash Drive performance with multiple software configurations



Product Description

The Flash Drive (iATAKIT) and the Flash Card (S2S2PLUSKIT) evaluation kits are designed to make DOS-based AT systems Intel Flash Card or Flash Drive-ready — quickly and simply. Each kit provides everything necessary to add a PCMCIA socket to a desktop system.

The Flash Drive and Flash Card kits are ideally suited for designers creating PDA-like products; designers of mobile, handheld and sub-notebook products; designers who want to evaluate Flash Card and/or Flash Drive products for embedded applications; and ISVs who write software for Flash Card or Flash Drive products.

To shorten time to market, both kits contain all necessary tools in a single box. Thus eliminating the time-consuming task of seeking and obtaining separate components.

Each kit includes SystemSoft's CardSoft suite utilizing industry-standard PCMCIA software — as well as SystemSoft's Flash File System (FFS) software. Additionally, each kit includes low-level driver and MTD source code, enabling modification for your specific requirements or product differentiation. For easy conversion of PCMCIA-ATA to IDE pinouts a 68-pin to 40/44-pin adapter is included.

intel

Features	Benefits
— All-in-one kit	— Contains everything needed to develop Intel Flash Card or Flash Drive-ready systems
— Includes low-level driver and MTD source code	— Enables code modification for specific requirements
— SystemSoft CardSoft	— Industry-standard PCMCIA software suite
— Flash File System (FFS) software	— Industry-standard file system from SystemSoft
— 68-pin ATA to 40/44-pin IDE adapter	— Easily converts PCMCIA-ATA pinout to IDE

The Intel logo is a registered trademark of Intel Corporation.
Other brands and names are the property of their respective owners.

Printed in USA/0694/40K/ASI/LK
©1994 Intel Corporation

Order Number: 297506-001

intel®

8

Flash Automotive Components

A28F400BR-T/B 4-MBIT (256K X 16, 512K X 8) SmartVoltage BOOT BLOCK FLASH MEMORY FAMILY

Automotive

- Intel SmartVoltage Technology
  - 5V or 12V Program/Erase
  - 5V Read Operation
- Very High Performance Read
  - 80 ns Max. Access Time,
  - −40 ns Max. Output Enable T
- Low Power Consumption
  - Maximum 60 mA Read Current at 5V
- x8/x16-Selectable Input/Output Bus
  - High Performance 16- or 32-bit CPUs
- Optimized Array Blocking Architecture
  - One 16-KB Protected Boot Block
  - Two 8-KB Parameter Blocks
  - One 96-KB Main Block
  - Three 128-KB Main Blocks
  - Top or Bottom Boot Locations
- Hardware-Protection for Boot Block
- Software EEPROM Emulation with Parameter Blocks
- Automotive Temperature Operation
  - −40°C to +125°C
- Extended Cycling Capability
  - 10,000 Block Erase Cycles for Parameter Blocks
  - 1,000 Block Erase Cycles for Main Blocks
- Automated Word/Byte Write and Block Erase
  - Industry-Standard Command User Interface
  - Status Registers
  - Erase Suspend Capability
- SRAM-Compatible Write Interface
- Automatic Power Savings Feature
  - 1 mA Typical I_{CC} Active Current in Static Operation
- Reset/Deep Power-Down Input
  - 0.2 μA I_{CC} Typical
  - Provides Reset for Boot Operations
- Hardware Data Protection Feature
  - Erase/Write Lockout during Power Transitions
- Industry-Standard Surface Mount Packaging
  - 44-Lead PSOP: JEDEC ROM Compatible
- ETOX™ IV Flash Technology

8

December 1994
Order Number: 290538-001

A28F400BR-T/B 4-MBIT (256K X 16, 512K X 8) SmartVoltage BOOT BLOCK FLASH MEMORY FAMILY

CONTENTS

PAGE

1.0 PRODUCT FAMILY OVERVIEW 8-3
- 1.1 New Features in the SmartVoltage Products 8-3
- 1.2 Main Features 8-3
- 1.3 Applications 8-4
- 1.4 Pinouts 8-4
- 1.5 Pin Descriptions 8-6

2.0 PRODUCT DESCRIPTON 8-7
- 2.1 Memory Organization 8-7
  - 2.1.1 BLOCKING 8-7
    - 2.1.1.1 Boot Block - 1 x 16 KB ... 8-7
    - 2.1.1.2 Parameter Blocks - 2 x 8 KB 8-8
    - 2.1.1.3 Main Blocks - 1 x 96 KB + 3 x 128 KB 8-8

3.0 PRODUCT FAMILY PRINCIPLES OF OPERATION 8-8
- 3.1 Bus Operations 8-10
- 3.2 Read Operations 8-10
  - 3.2.1 READ ARRAY 8-10
    - 3.2.1.1 Output Control 8-10
    - 3.2.1.2 Input Control 8-10
  - 3.2.2 INTELLIGENT IDENTIFIERS 8-10
- 3.3 Write Operations 8-10
  - 3.3.1 COMMAND USER INTERFACE 8-10
    - 3.3.1.1 Command Function Description 8-11
  - 3.3.2 STATUS REGISTER 8-13
    - 3.3.2.1 Clearing the Status Register 8-13
  - 3.3.3 PROGRAM MODE 8-13
  - 3.3.4 ERASE MODE 8-15
    - 3.3.4.1 Suspending and Resuming Erase 8-15

CONTENTS

PAGE

- 3.4 Boot Block Locking 8-19
  - 3.4.1 $V_{PP} = V_{IL}$ FOR COMPLETE PROTECTION 8-19
  - 3.4.2 WP# $= V_{IL}$ FOR BOOT BLOCK LOCKING 8-19
  - 3.4.3 RP# $= V_{HH}$ OR WP# $= V_{IH}$ FOR BOOT BLOCK UNLOCKING 8-19
- 3.5 Power Consumption 8-19
  - 3.5.1 ACTIVE POWER 8-19
  - 3.5.2 AUTOMATIC POWER SAVINGS 8-19
  - 3.5.3 STANDBY POWER 8-19
  - 3.5.4 DEEP POWER-DOWN MODE 8-19
- 3.6 Power-Up Operation 8-20
  - 3.6.1 RP# CONNECTED TO SYSTEM RESET 8-20
- 3.7 Power Supply Decoupling 8-20
  - 3.7.1 V_{PP} TRACE ON PRINTED CIRCUIT BOARDS 8-20
  - 3.7.2 V_{CC}, V_{PP} AND RP# TRANSITIONS 8-20

4.0 ABSOLUTE MAXIMUM RATINGS .. 8-21

5.0 OPERATING CONDITIONS 8-22
- 5.1 V_{CC} Voltage 8-22
- 5.2 DC Characteristics 8-23
- 5.3 AC Characteristics 8-27

6.0 ADDITIONAL INFORMATION 8-33
- 6.1 Ordering Information 8-33
- 6.2 References 8-33
- 6.3 Revision History 8-34

1.0 PRODUCT FAMILY OVERVIEW

This datasheet contains the specifications for the automotive version of the 28F400BR family of boot block flash memory devices.

This device continues to offer the same functionality as earlier "BX" devices but adds the capability of performing program and erase operations with a 5V or 12V V_{PP}. Furthermore, the A28F400BR automatically senses which voltage is applied to the V_{PP} pin and adjusts its operation accordingly

1.1 New Features in the SmartVoltage Products

The new SmartVoltage boot block flash memory family offers identical operation as the current BX/BL 12V program products, except for the differences listed below. All other functions are equivalent to current products, including signatures, write commands, and pinouts.

- WP# pin has replaced a DU pin. See Table 1 for details.
- 5V Program/Erase operation has been added that uses proven program and erase techniques with 5V ± 10% applied to V_{PP}.

If you are designing with existing "BX" 12V V_{PP} boot block products today, you should provide the capability in your board design to upgrade to these new SmartVoltage products.

Follow these guidelines to ensure compatibilty:

1. Connect WP# (DU on existing products) to a control signal, V_{CC} or GND.
2. If adding a switch on V_{PP} for write protection, switch to GND for complete write protection.
3. Allow for connecting 5V to V_{PP} instead of 12V, if desired.

1.2 Main Features

Intel's SmartVoltage technology provides the most flexible voltage solution in the industry. SmartVoltage provides two discrete voltage supply pins, V_{CC} for read operation, and V_{PP} for Program and Erase operation. Discrete supply pins allow system designers to use the optimal voltage levels for their design. For program and erase operations, 5V V_{PP} operation eliminates the need for in system voltage converters, while 12V V_{PP} operation provides faster program and erase for situations where 12V is available, such as manufacturing or designs where 12V is already available.

The 28F400 boot block flash memory family is a very high-performance, 4-Mbit (4,194,304 bit) flash memory family organized as either 256 Kwords (262,144 words) of 16 bits each or 512 Kbytes (524,288 bytes) of 8 bits each.

Separately erasable blocks, including a hardware-lockable boot block (16,384 bytes), two parameter blocks (8,192 Bytes each) and main blocks (one block of 98,304 bytes and three blocks of 131,072 bytes) define the boot block flash family architecture. See Figure 3 for memory maps. Each parameter block can be independently erased and programmed 10,000 times. Each main block can be erased 1,000 times.

The boot block is located at either the top (denoted by -T suffix) or the bottom (-B suffix) of the address map in order to accommodate different microprocessor protocols for boot code location. The hardware-lockable boot block provides complete code security for the kernel code required for system initialization. Locking and unlocking of the boot block is controlled by WP# and/or RP# (see Section 3.4 for details).

The Command User Interface (CUI) serves as the interface between the microprocessor or microcontroller and the internal operation of the boot block flash memory products. The internal Write State Machine (WSM) automatically executes the algorithms and timings necessary for program and erase operations, including verifications, thereby unburdening the microprocessor or microcontroller of these tasks. The Status Register (SR) indicates the status of the WSM and whether it successfully completed the desired program or erase operation.

Program and Erase Automation allows program and Erase operations to be executed using an industry-standard two-write command sequence to the CUI. Data writes are performed in word or byte increments. Each byte or word in the flash memory can be programmed independently of other memory locations, unlike erases, which erase all locations within a block simultaneously.

The 4-Mbit SmartVoltage boot block flash memory family is also designed with an Automatic Power Savings (APS) feature which minimizes system battery current drain, allowing for very low power designs. To provide even greater power savings, the boot block family includes a deep power-down mode which minimizes power consumption by turning most of the Flash memory's circuitry off. This mode is controlled by the RP# pin and its usage is discussed in Section 3.5, along with other power consumption issues.

Additionally, the RP# pin provides protection against unwanted command writes due to invalid system bus conditions that may occur during system reset and power-up/down sequences. Also, when the Flash memory powers-up, it automatically defaults to the read array mode, but during a warm system reset, where power continues uniterrupted to the system components, the flash memory could remain in a non-read mode, such as erase. Consequently, the system Reset pin should be tied to RP# to reset the memory to normal read mode upon activation of the Reset pin.

The byte-wide or word-wide input/output is controlled by the BYTE# pin. See Table 1 for a detailed description of BYTE# operations, especially the usage of the DQ_{15}/A_{-1} pin.

The 28F400 products are available in a ROM/EPROM-compatible pinout and housed in the 44-lead PSOP (Plastic Small Outline) package.

Refer to the DC Characteristics Table Section 5.2 for complete current and voltage specifications. Refer to the AC Characteristics Table, Section 5.3, for read, write and erase performance specifications.

1.3 Applications

The 4-Mbit boot block flash memory family combines high-density, low-power, high-performance, cost-effective flash memories with blocking and hardware protection capabilities. Their flexibility and versatility reduce costs throughout the product life cycle. Flash memory is ideal for Just-In-Time production flow, reducing system inventory and costs, and eliminating component handling during the production phase.

When the product is in the end-user's hands, and updates or feature enhancements become necessary or mandatory, flash memory eliminates the need to replace an assembly. The update can be performed as part of routine maintenance operation by relatively unsophisticated technicians.

The reliability of such a field upgrade is enhanced by a hardware-protected 16-Kbyte boot block. If the protection methods are implemented in the circuit design, the boot block will be unchangeable. Locating the boot-strap code in this area assures a fail-safe recovery from an update operation that failed to complete correctly.

The two 8-Kbyte parameter blocks allow modification of control algorithms to reflect changes in the process or device being controlled. A variety of software algorithms allow these two blocks to behave like a standard EEPROM.

Intel's boot block architecture provides a flexible voltage solution for the different design needs of various applications. The asymmetrically blocked memory map allows the integration of several memory components into a single Flash device. The boot block provides a secure boot PROM; the parameter blocks can emulate EEPROM functionality for parameter store with proper software techniques; and the main blocks provide code and data storage with access times fast enough to execute code in place, decreasing RAM requirements.

1.4 Pinouts

Intel's SmartVoltage Boot Block architecture provides upgrade paths in every package pinout to the 8-Mbit density. The 28F400 44-lead PSOP pinout follows the industry standard ROM/EPROM pinout as shown in Figure 2.

Pinouts for the corresponding 2-Mbit and 8-Mbit components are also provided for convenient reference. 4-Mbit pinouts are given on the chip illustration in the center, with 2-Mbit and 8-Mbit pinouts going outward from the center.

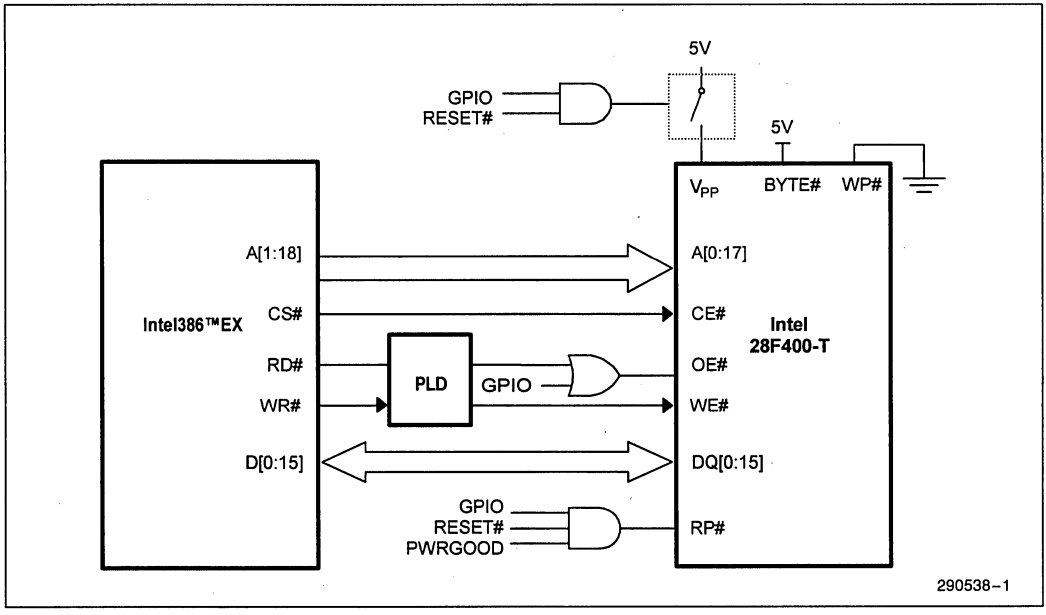
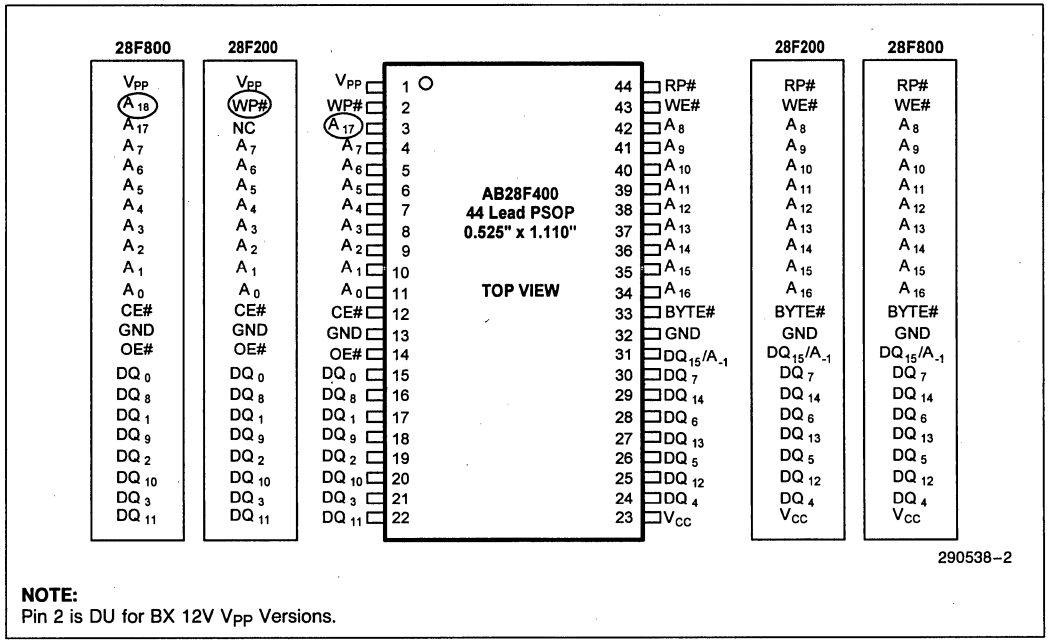


Figure 1. 28F400BX Interface to Intel386™ Microprocessor



NOTE:
Pin 2 is DU for BX 12V V_{PP} Versions.

Figure 2. 44-Lead PSOP Lead Configuration for x8/x16 28F400 Is Compatible with 2 and 8 Mbit.

8

1.5 Pin Descriptions

Table 1. 28F400 Pin Descriptions

Symbol	Type	Name and Function
A_0–A_{17}	INPUT	**ADDRESS INPUTS:** for memory addresses. Addresses are internally latched during a write cycle.
A_9	INPUT	**ADDRESS INPUT:** When A_9 is at V_{HH} the signature mode is accessed. During this mode, A_0 decodes between the manufacturer and device IDs. When BYTE# is at a logic low, only the lower byte of the signatures are read. DQ_{15}/A_{-1} is a don't care in the signature mode when BYTE# is low.
DQ_0–DQ_7	INPUT/OUTPUT	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a Program command. Inputs commands to the Command User Interface when CE# and WE# are active. Data is internally latched during the Write cycle. Outputs array, Intelligent Identifier and Status Register data. The data pins float to tri-state when the chip is de-selected or the outputs are disabled.
DQ_8–DQ_{15}	INPUT/OUTPUT	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a Program command. Data is internally latched during the Write cycle. Outputs array data. The data pins float to tri-state when the chip is de-selected or the outputs are disabled as in the byte-wide mode (BYTE# = "0"). In the byte-wide mode DQ_{15}/A_{-1} becomes the lowest order address for data output on DQ_0–DQ_7.
CE#	INPUT	**CHIP ENABLE:** Activates the device's control logic, input buffers, decoders and sense amplifiers. CE# is active low. CE# high de-selects the memory device and reduces power consumption to standby levels. If CE# and RP# are high, but not at a CMOS high level, the standby current will increase due to current flow through the CE# and RP# input stages.
OE#	INPUT	**OUTPUT ENABLE:** Enables the device's outputs through the data buffers during a read cycle. OE# is active low.
WE#	INPUT	**WRITE ENABLE:** Controls writes to the Command Register and array blocks. WE# is active low. Addresses and data are latched on the rising edge of the WE# pulse.
RP#	INPUT	**RESET/DEEP POWER-DOWN:** Uses three voltage levels (V_{IL}, V_{IH}, and V_{HH}) to control two different functions: reset/deep power-down mode and boot block unlocking. It is backwards-compatible with the 28F400BX/BL. **When RP# is at logic low, the device is in reset/deep power-down mode,** which puts the outputs at High-Z, resets the Write State Machine, and draws minimum current. **When RP# is at logic high, the device is in standard operation.** When RP# transitions from logic-low to logic-high, the device defaults to the read array mode. **When RP# is at V_{HH}, the boot block is unlocked** and can be programmed or erased. This overides any control from the WP# input.

Table 1. 28F400 Pin Descriptions (Continued)

Symbol	Type	Name and Function
WP#	INPUT	**WRITE PROTECT:** Provides a method for unlocking the boot block in a system without a 12V supply. **When WP# is at logic low, the boot block is locked,** preventing Program and Erase operations to the boot block. If a Program or Erase operation is attempted on the boot block when WP# is low, the corresponding status bit (bit 4 for Program, bit 5 for Erase) will be set in the Status Register to indicate the operation failed. **When WP# is at logic high, the boot block is unlocked** and can be programmed or erased. **NOTE:** This feature is overridden and the boot block unlocked when RP# is at V_{HH}. See Section 3.4 for details on write protection.
BYTE#	INPUT	**BYTE# ENABLE:** Controls whether the device operates in the byte-wide (x8) mode or the word (x16) mode. The BYTE# input must be controlled at CMOS levels to meet the CMOS current specification in the standby mode. When BYTE# is at logic low, the byte-wide mode is enabled. A 19 bit address is applied on A_{-1} to A_{17}, and 8 bits of data is read and written on DQ_0–DQ_7. When BYTE# is at logic high, the word-wide mode is enable. An 18 bit address is applied on A_0 to A_{17} and 16 bits of data is read and written on DQ_0–DQ_{15}.
V_{CC}		**DEVICE POWER SUPPLY:** 5.0V ± 10%
V_{PP}		**PROGRAM/ERASE POWER SUPPLY:** For erasing memory array blocks or programming data in each block, a voltage either of 5V ± 10% or 12V ± 5% must be applied to this pin. When $V_{PP} < V_{PPLK}$ all blocks are locked and protected against Program and Erase commands.
GND		**GROUND:** For all internal circuitry.
NC		**NO CONNECT:** Pin may be driven or left floating.

2.0 PRODUCT DESCRIPTON

2.1 Memory Organization

2.1.1 BLOCKING

This product family features an asymmetrically blocked architecture enhancing system memory integration. Each block can be erased independently of the others up to 10,000 times. The block sizes have been chosen to optimize their functionality for common applications of nonvolatile storage. For the address locations of the blocks, see the memory maps in Figure 3.

2.1.1.1 Boot Block - 1 x 16 KB

The boot block is intended to replace a dedicated boot PROM in a microprocessor or microcontroller-based system. The 16-Kbyte (16,384 bytes) boot block is located at either the top (denoted by -T suffix) or the bottom (-B suffix) of the address map to accommodate different microprocessor protocols for boot code location. This boot block features hardware controllable write-protection to protect the crucial microprocessor boot code from accidental erasure. The protection of the boot block is controlled using a combination of the V_{PP}, RP#, and WP# pins, as is detailed in Table 8.

8

2.1.1.2 Parameter Blocks - 2 x 8 KB

The boot block architecture includes parameter blocks to facilitate storage of frequently updated small parameters that would normally require an EEPROM. By using software techniques, the byte-rewrite functionality of EEPROMs can be emulated. These techniques are detailed in Intel's AP-604, "Using Intel's Boot Block Flash Memory Parameter Blocks to Replace EEPROM." Each boot block component contains two parameter blocks of eight Kbytes (8,192 bytes) each. The parameter blocks are not write-protectable.

2.1.1.3 Main Blocks - 1 x 96 KB + 3 x 128 KB

After the allocation of address space to the boot and parameter blocks, the remainder is divided into main blocks for data or code storage. Each 4-Mbit device contains one 96-Kbyte (98,304 byte) block and three 128-Kbyte (131,072 byte) blocks. See the memory maps for each device for more information.

3.0 PRODUCT FAMILY PRINCIPLES OF OPERATION

Flash memory augments EPROM functionality with in-circuit electrical write and erase. The boot block flash family utilizes a Command User Interface (CUI) and automated algorithms to simplify write and erase operations. The CUI allows for 100% TTL-level control inputs, fixed power supplies during erasure and programming, and maximum EPROM compatibility.

When $V_{PP} < V_{PPLK}$, the device will only successfully execute the following commands: Read Array, Read Status Register, Clear Status Register and intelligent identifier mode. The device provides standard EPROM Read, Standby and Output Disable operations. Manufacturer Identification and Device Identification data can be accessed through the CUI or through the standard EPROM A_9 high voltage access (V_{ID}) for PROM programming equipment.

The same EPROM Read, Standby and Output Disable functions are available when 5V or 12V is applied to the V_{PP} pin. In addition, 5V or 12V on V_{PP} allows write and erase of the device. All functions associated with altering memory contents: Write and Erase, Intelligent Identifier Read, and Read Status are accessed via the CUI.

The purpose of the Write State Machine (WSM) is to completely automate the write and erasure of the device. The WSM will begin operation upon receipt of a signal from the CUI and will report status back through a Status Register. The CUI will handle the WE# interface to the data and address latches, as well as system software requests for status while the WSM is in operation.

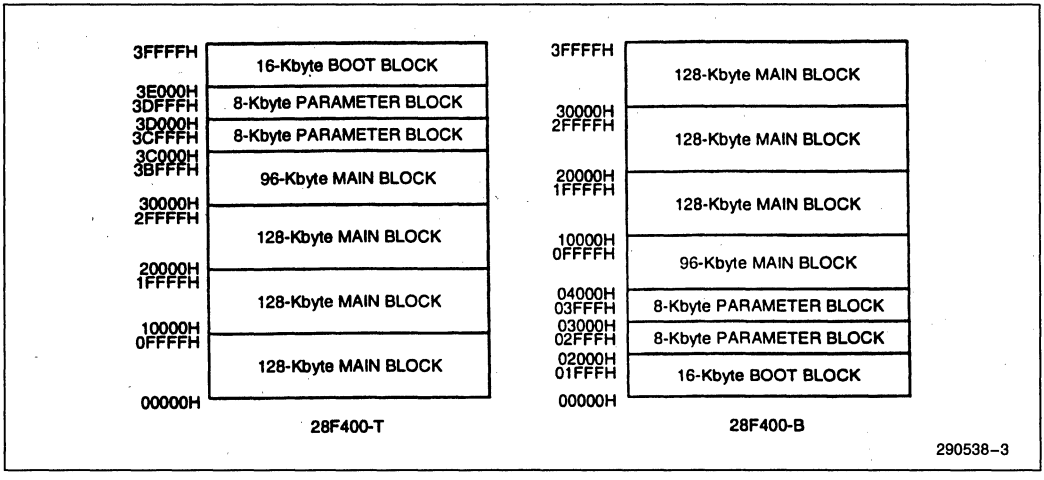


Figure 3. 28F400-T/B Memory Maps

Table 2. Bus Operations for Word-Wide Mode (BYTE# = V_{IH})

Mode	Notes	RP#	CE#	OE#	WE#	A_9	A_0	V_{PP}	DQ_{0-15}
Read	1,2,3	V_{IH}	V_{IL}	V_{IL}	V_{IH}	X	X	X	D_{OUT}
Output Disable		V_{IH}	V_{IL}	V_{IH}	V_{IH}	X	X	X	High Z
Standby		V_{IH}	V_{IH}	X	X	X	X	X	High Z
Deep Power-Down	9	V_{IL}	X	X	X	X	X	X	High Z
Intelligent Identifier (Mfr)	4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IL}	X	0089 H
Intelligent Identifier (Device)	4,5	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IH}	X	See Table 4
Write	6,7,8	V_{IH}	V_{IL}	V_{IH}	V_{IL}	X	X	X	D_{IN}

Table 3. Bus Operations for Byte-Wide Mode (BYTE# = V_{IL})

Mode	Notes	RP#	CE#	OE#	WE#	A_9	A_0	A_{-1}	V_{PP}	DQ_{0-7}	DQ_{8-14}
Read	1,2,3	V_{IH}	V_{IL}	V_{IL}	V_{IH}	X	X	X	X	D_{OUT}	High Z
Output Disable		V_{IH}	V_{IL}	V_{IH}	V_{IH}	X	X	X	X	High Z	High Z
Standby		V_{IH}	V_{IH}	X	X	X	X	X	X	High Z	High Z
Deep Power-Down	9	V_{IL}	X	X	X	X	X	X	X	High Z	High Z
Intelligent Identifier (Mfr)	4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IL}	X	X	89H	High Z
Intelligent Identifier (Device)	4,5	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IH}	X	X	See Table 4	High Z
Write	6,7,8	V_{IH}	V_{IL}	V_{IH}	V_{IL}	X	X	X	X	D_{IN}	High Z

NOTES:
1. Refer to DC Characteristics.
2. X can be V_{IL}, V_{IH} for control pins and addresses, V_{PPLK} or V_{PPH} for V_{PP}.
3. See DC Characteristics for V_{PPLK}, $V_{PPH}1$, $V_{PPH}2$, V_{HH}, V_{ID} voltages.
4. Manufacturer and Device codes may also be accessed via a CUI write sequence, A_1–A_{17} = X, A_1–A_{18} = X.
5. See Table 4 of Device IDs.
6. Refer to Table 5 for valid D_{IN} during a Write operation.
7. Command writes for Block Erase or Word/Byte Write are only executed when V_{PP} = $V_{PPH}1$ or $V_{PPH}2$.
8. To Write or Erase the boot block, hold RP# at V_{HH} or WP# at V_{IH}.
9. RP# must be at GND ± 0.2V to meet the maximum deep power-down current specified.

8

3.1 Bus Operations

Flash memory reads, erases and writes in-system via the local CPU. All bus cycles to or from the flash memory conform to standard microprocessor bus cycles. These bus operations are summarized in Tables 2 and 3.

3.2 Read Operations

The boot block flash device has three user read modes: array, intelligent identifier, and status register. Status register read mode will be discussed, in detail, in Section 3.3.2.

3.2.1 READ ARRAY

When RP# transitions from V_{IL} (reset) to V_{IH}, the device will be in the read array mode and will respond to the read control inputs (CE#, address inputs, and OE#) without any commands being written to the CUI.

When the device is in the read array mode, five control signals must be controlled to obtain data at the outputs.

- WE# must be logic high (V_{IH})
- CE# must be logic low (V_{IL})
- OE must be logic low (V_{IL})
- RP# must be logic high (V_{IH})
- BYTE# must be logic high or logic low.

In addition, the address of the desired location must be applied to the address pins. Refer to Figure 10 and 11 for the exact sequence and timing of these signals.

If the device is not in read array mode, as would be the case after a Program or Erase operation, the Read Mode command (FFH) must be written to the CUI before reads can take place.

3.2.1.1 Output Control

With OE# at logic-high level (V_{IH}), the output from the device is disabled and data Input/Output pins (DQ[0:15] or DQ[0:7]) are tri-stated.

3.2.1.2 Input Control

With WE# at logic-high level (V_{IH}), input to the device is disabled.

3.2.2 INTELLIGENT IDENTIFIERS

The intelligent identifiers of the SmartVoltage boot block components are identical to the boot block products that operate only at 12V V_{PP}. The manufacturer and device codes are read via the CUI or by taking the A_9 pin to V_{ID}. Writing 90H to the CUI places the device into intelligent identifier read mode. In this mode, $A_0 = 0$ outputs the manufacturer's identification code and $A_0 = 1$ outputs the device code. When BYTE# is at a logic low, only the lower byte of the above signatures is read and DQ_{15}/A_{-1} is a "don't care" during intelligent identifier mode. See the table below for product signatures. A Read Array command must be written to the memory to return to the read array mode.

Table 4. Intelligent Identifier Table

Product	Mfr. ID	Device ID	
		-T (Top Boot)	-B (Bottom Boot)
28F400	0089 H	4470 H	4471 H

3.3 Write Operations

3.3.1 COMMAND USER INTERFACE (CUI)

The Command User Interface (CUI) serves as the interface between the microprocessor and the internal chip controller. Commands are written to the CUI using standard microprocessor write timings. The available commands are Read Array, Read Intelligent Identifier, Read Status Register, Clear Status Register, Erase and Program (summarized in Tables 5 and 6). For Read commands, the CUI points the read path at either the array, the intelligent identifier, or the Status Register depending on the command received. For Program or Erase commands, the CUI informs the Write State Machine (WSM) that a write or erase has been requested. During the execution of a Program command, the WSM will control the programming sequences and the CUI will only respond to status reads. During an erase cycle, the CUI will respond to status reads and erase

Table 5. Command Set Codes and Corresponding Device Mode

Command Codes	Device Mode
00	Invalid Reserved
10	Alternate Program Set-Up
20	Erase Set-Up
40	Program Set-Up
50	Clear Status Register
70	Read Status Register
90	Intelligent Identifier
B0	Erase Suspend
D0	Erase Resume/ Erase Confirm
FF	Read Array

suspend. After the WSM has completed its task, it will set the WSM Status bit to a "1", which will also allow the CUI to respond to its full command set. Note that after the WSM has returned control to the CUI, the CUI will stay in the current command state until it receives another command.

3.3.1.1 Command Function Description

Device operations are selected by writing specific commands into the CUI. Table 5 defines the available commands.

Invalid/Reserved

These are unassigned commands and should not be used. Intel reserves the right to redefine these codes for future functions.

Read Array (FFH)

This single write cycle command points the read path at the array. If the host CPU performs a CE#/OE#-controlled Read immediately following a two-write sequence that started the WSM, then the device will output Status Register contents. If the Read Array command is given after the Erase Setup command, the device will reset to read the array. A two Read Array command sequence (FFH) is required to reset to Read Array after the Program Set-up command.

Intelligent Identifier (90H)

After this command is executed, the CUI points the output path to the intelligent identifier circuits. Only intelligent identifier values at addresses 0 and 1 can be read (only address A_0 is used in this mode, all other address inputs are ignored).

Read Status Register (70H)

This is one of the two commands that is executable while the WSM is operating. After this command is written, a read of the device will output the contents of the Status Register, regardless of the address presented to the device.

The device automatically enters this mode after program or erase has completed.

Clear Status Register (50H)

The WSM can only set the Program Status and Erase Status bits in the Status Register to "1", it cannot clear them to "0".

Two reasons exist for operating the Status Register in this fashion. The first is synchronization. Since the WSM does not know when the host CPU has read the Status Register, it would not know when to clear the status bits. Secondly, if the CPU is programming a string of bytes, it may be more efficient to query the Status Register after programming the string. Thus, if any errors exist while programming the string, the Status Register will return the accumulated error status.

Table 6. Command Bus Definitions

Command	Notes	First Bus Cycle			Second Bus Cycle		
	8	Oper	Addr	Data	Oper	Addr	Data
Read Array	1	Write	X	FFH			
Intelligent Identifier	2,4	Write	X	90H	Read	IA	IID
Read Status Register	3	Write	X	70H	Read	X	SRD
Clear Status Register		Write	X	50H			
Word/Byte Write	6,7	Write	WA	40H	Write	WA	WD
Alternate Word/Byte Write	6,7	Write	WA	10H	Write	WA	WD
Block Erase/Confirm	5	Write	BA	20H	Write	BA	D0H
Erase Suspend/Resume		Write	X	B0H	Write	X	D0H

ADDRESS
BA= Block Address
IA= Identifier Address
WA= Write Address
X= Don't Care

DATA
SRD= Status Register Data
IID= Identifier Data
WD= Write Data

NOTES:
1. Bus operations are defined in Tables 2 and 3.
2. IA = Identifier Address: $A_0=0$ for manufacturer code, $A_0=1$ for device code.
3. SRD – Data read from Status Register.
4. IID = Intelligent Identifier Data. Following the Intelligent Identifier command, two Read operations access manufacturer and device codes.
5. BA = Address within the block being erased.
6. WA = Address to be written. WD = Data to be written at location WD.
7. Either 40H or 10H commands is valid.
8. When writing commands to the device, the upper data bus [DQ_8–DQ_{15}] = X (28F400 only) which is either V_{CC} or V_{SS}, to minimize current draw.

Program Setup (40H or 10H)

This command simply sets the CUI into a state such that the next write will load the Address and Data registers. After this command is executed, the outputs default to the Status Register. A two Read Array command sequence (FFH) is required to reset to Read Array after the Program Setup command.

Program

The second write after the Program Setup command, will latch addresses and data. Also, the CUI initiates the WSM to begin execution of the program algorithm. The device outputs Status Register data when OE# is enabled. A Read Array command is required after programming, to read array data.

Erase Setup (20H)

Prepares the CUI for the Erase Confirm command. No other action is taken. If the next command is not an Erase Confirm command, then the CUI will set both the Program Status and Erase Status bits of the Status Register to a "1", place the device into the Read Status Register state, and wait for another command.

Erase Confirm (D0H)

If the previous command was an Erase Setup command, then the CUI will enable the WSM to erase, at the same time closing the address and data latches, and respond only to the Read Status Register and Erase Suspend commands. While the WSM is executing, the device will output Status Register data when OE# is toggled low. Status Register data can only be updated by toggling either OE# or CE# low.

Erase Suspend (B0H)

This command is only valid while the WSM is executing an Erase operation, and therefore will only be responded to during an Erase operation. After this command has been executed, the CUI will set an output that directs the WSM to suspend Erase operations, and then respond only to Read Status Register or to the Erase Resume commands. Once the WSM has reached the Suspend state, it will set an output into the CUI which allows the CUI to respond to the Read Array, Read Status Register, and Erase Resume commands. In this mode, the CUI will not respond to any other commands. The WSM will also set the WSM Status bit to a "1". The WSM will continue to run, idling in the SUSPEND state, regardless of the state of all input control pins except RP#, which will immediately shut down the WSM and the remainder of the chip, if it is made active. During a Suspend operation, the data and address latches will remain closed, but the address pads are able to drive the address into the read path.

Erase Resume (D0H)

This command will cause the CUI to clear the Suspend state and clear the WSM Status Bit to a "0", but only if an Erase Suspend command was previously issued. Erase Resume will not have any effect under any other conditions.

3.3.2 STATUS REGISTER

The device contains a Status Register which may be read to determine when a Program or Erase operation is complete, and whether that operation completed successfully. The Status Register may be read at any time by writing the Read Status command to the CUI. After writing this command, all subsequent Read operations output data from the Status Register until another command is written to the CUI. A Read Array command must be written to the CUI to return to the read array mode.

The Status Register bits are output on DQ[0:7], whether the device is in the byte-wide (x8) or word-wide (x16) mode. In the word-wide mode the upper byte, DQ[8:15], is set to 00H during a Read Status command. In the byte-wide mode, DQ[8:14] are tri-stated and DQ_{15}/A_{-1} retains the low order address function.

Important: *The contents of the Status Register are latched on the falling edge of OE# or CE#, whichever occurs last in the read cycle.* This prevents possible bus errors which might occur if the contents of the Status Register change while reading the Status Register. CE# or OE# must be toggled with each subsequent status read, or the completion of a Program or Erase operation will not be evident from the Status Register.

When the WSM is active, this register will indicate the status of the WSM, and will also hold the bits indicating whether or not the WSM was successful in performing the desired operation.

3.3.2.1 Clearing the Status Register

The WSM sets status bits "3" through "7" to "1", and clears bits "6" and "7" to "0", but cannot clear status bits "3" through "5" to "0". Bits 3 through 5 can only be cleared by the controlling CPU through the use of the Clear Status Register command. These bits can indicate various error conditions. By allowing the system software to control the resetting of these bits, several operations may be performed (such as cumulatively programming several bytes or erasing multiple blocks in sequence). The Status Register may then be read to determine if an error occurred during that programming or erasure series. This adds flexibility to the way the device may be programmed or erased. To clear the Status Register, the Clear Status Register command is written to the CUI. Then, any other command may be issued to the CUI. Note, again, that before a read cycle can be initiated, a Read Array command must be written to the CUI to specify whether the read data is to come from the Memory Array, Status Register, or Intelligent Identifier.

3.3.3 PROGRAM MODE

Programing is executed using a two-write sequence. The Program Setup command is written to the CUI followed by a second write which specifies the

address and data to be programmed. The WSM will execute a sequence of internally timed events to:

1. Program the desired bits of the addressed memory word or byte.
2. Verify that the desired bits are sufficiently programmed.

Programming of the memory results in specific bits within a byte or word being changed to a "0".

If the user attempts to program "1"s, there will be no change of the memory cell content and no error occurs.

Similar to erasure, the Status Register indicates whether programming is complete. While the program sequence is executing, bit 7 of the Status Register is a "0". The Status Register can be polled by toggling either CE# or OE# to determine when the program sequence is complete. Only the Read Status Register command is valid while programming is active.

Table 7. Status Register Bit Definition

WSMS	ESS	ES	DWS	VPPS	R	R	R
7	6	5	4	3	2	1	0

	NOTES:
SR.7 = WRITE STATE MACHINE STATUS (WSMS) 1 = Ready 0 = Busy	Write State Machine bit must first be checked to determine Byte/Word program or Block Erase completion, before the Program or Erase Status bits are checked for success.
SR.6 = ERASE-SUSPEND STATUS (ESS) 1 = Erase Suspended 0 = Erase In Progress/Completed	When Erase Suspend is issued, WSM halts execution and sets both WSMS and ESS bits to "1". ESS bit remains set to "1" until an Erase Resume command is issued.
SR.5 = ERASE STATUS 1 = Error In Block Erasure 0 = Successful Block Erase	When this bit is set to "1", WSM has applied the maximum number of erase pulses to the block and is still unable to successfully verify block erasure.
SR.4 = PROGRAM STATUS 1 = Error in Byte/Word Program 0 = Successful Byte/Word Program	When this bit is set to "1", WSM has attempted but failed to program a byte or word.
SR.3 = V_{PP} STATUS 1 = V_{PP} Low Detect, Operation Abort 0 = V_{PP} OK	The V_{PP} Status bit, unlike an A/D converter, does not provide continuous indication of V_{PP} level. The WSM interrogates V_{PP} level only after the Byte Write or Erase command sequences have been entered, and informs the system if V_{PP} has not been switched on. The V_{PP} Status bit is not guaranteed to report accurate feedback between V_{PPLK} and V_{PPH}.
SR.2–SR.0 = RESERVED FOR FUTURE ENHANCEMENTS	These bits are reserved for future use and should be masked out when polling the Status Register.

When programming is complete, the status bits, which indicate whether the Program operation was successful, should be checked. If bit 3 is set to a "1", then V_{PP} was not within acceptable limits, and the WSM did not execute the programming sequence. If the Program operation fails, Bit 4 of the Status Register will be set within 3.3 ms as determined by the timeout of the WSM.

The Status Register should be cleared before attempting the next operation. Any CUI instruction can follow after programming is completed; however, reads from the Memory Array, Status Register, or Intelligent Identifier cannot be accomplished until the CUI is given the Read Array command.

3.3.4 ERASE MODE

Erasure of a single block is initiated by writing the Erase Setup and Erase Confirm commands to the CUI, along with the addresses identifying the block to be erased. These addresses are latched internally when the Erase Confirm command is issued. Block erasure results in all bits within the block being set to "1".

The WSM will execute a sequence of internally timed events to:

1. Program all bits within the block to "0".
2. Verify that all bits within the block are sufficiently programmed to "0".
3. Erase all bits within the block.
4. Verify that all bits within the block are sufficiently erased.

While the erase sequence is executing, bit 7 of the Status Register is a "0".

When the Status Register indicates that erasure is complete, the status bits, which indicate whether the Erase operation was successful, should be checked. If the Erase operation was unsuccessful, bit 5 of the Status Register will be set to a "1", indicating an Erase Failure. If V_{PP} was not within acceptable limits after the Erase Confirm command is issued, the WSM will not execute an erase sequence; instead, bit 5 of the Status Register is set to a "1" to indicate an Erase Failure, and bit 3 is set to a "1" to identify that V_{PP} supply voltage was not within acceptable limits.

The Status Register should be cleared before attempting the next operation. Any CUI instruction can follow after erasure is completed; however, reads from the Memory Array, Status Register, or Intelligent Identifier cannot be accomplished until the CUI is given the Read Array command.

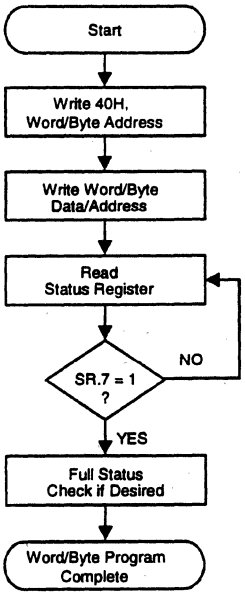
3.3.4.1 Suspending and Resuming Erase

Since an Erase operation requires on the order of seconds to complete, an Erase Suspend command is provided to allow erase-sequence interruption in order to read data from another block of the memory. Once the erase sequence is started, writing the Erase Suspend command to the CUI requests that the WSM pause the erase sequence at a pre-determined point in the erase algorithm. The Status Register must then be read to determine if the Erase operation has been suspended.

At this point, a Read Array command can be written to the CUI in order to read data from blocks other than that which is being suspended. The only other valid command at this time is the Erase Resume command or Read Status Register command.

During erase suspend mode, the chip can go into a pseudo-standby mode by taking CE# to V_{IH}, which reduces active current draw.

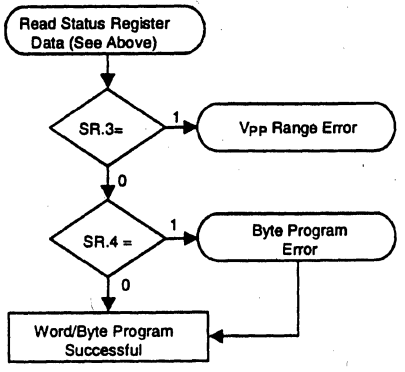
To resume the Erase operation, the chip must be enabled by taking CE# to V_{IL}, then issuing the Erase Resume command. When the Erase Resume command is given, the WSM will continue with the erase sequence and complete erasing the block. As with the end of a standard erase operation, the Status Register must be read, cleared, and the next instruction issued in order to continue.



Bus Operation	Command	Comments
Write	Setup Program	Data = 40H Addr = Word/Byte to Program
Write	Program	Data = Data to Program Addr = Location to Program
Read		Status Register Data Toggle CE# or OE# to Update SRD.
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy

Repeat for subsequent Word/Byte Writes.
SR Full Status Check can be done after each Word/Byte Write, or after a sequence of Word/Byte Writes.
Write FFH after the last write operation to reset device to read array mode.

FULL STATUS CHECK PROCEDURE



Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4 1 = V_{PP} Byte Program Error

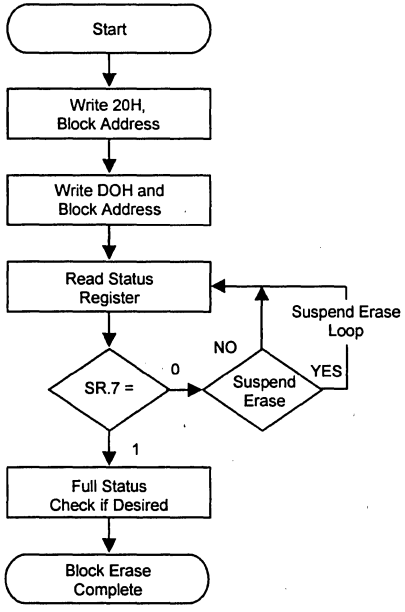
SR.3 MUST be cleared, if set during a program attempt, before further attempts are allowed by the Write State Machine.

SR.4 is only clear by the Clear Status Register Command, in cases where multiple bytes are programmed before full status is checked.

If error is detected, clear the Status Register before attempting retry or other error recovery.

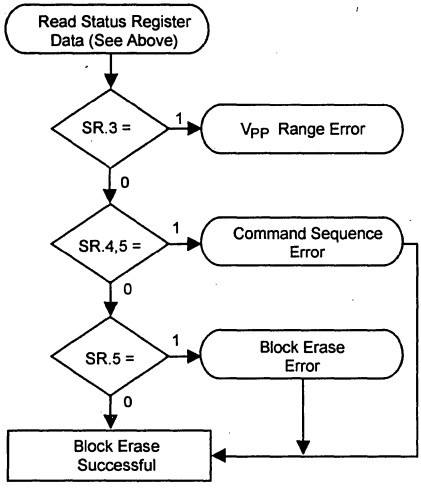
290538-4

Figure 4. Automated Word/Byte Programming Flowchart



Bus Operation	Command	Comments
Write	Erase Setup	Data = 20H Addr = Within Block to be Erased
Write	Erase Confirm	Data = D0H Addr = Within Block to be Erased
Read		Status Register Data Toggle CE# or OE# to Update Status Register
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy
Repeat for subsequent block erasures. Full Status Check can be done after each block erase, or after a sequence of block erasures. Write FFH after the last operation to reset device to read array mode.		

FULL STATUS CHECK PROCEDURE

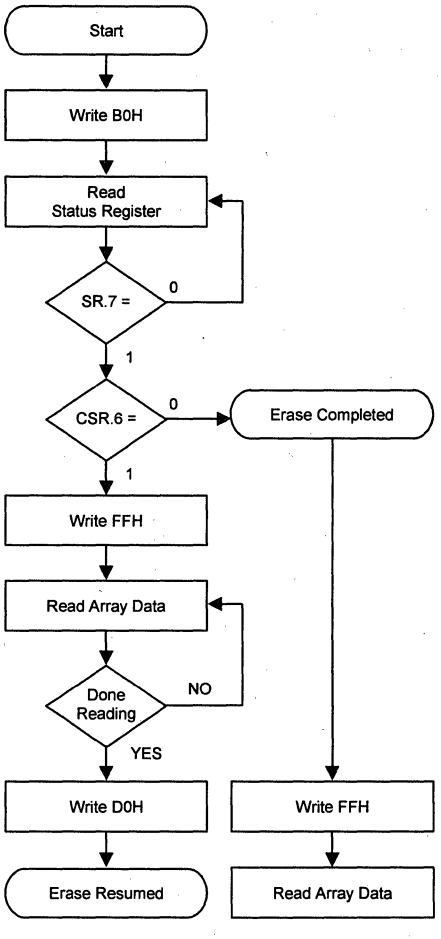


Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4,5 Both 1 = Command Sequence Error
Standby		Check SR.5 1 = Block Erase Error
SR.3 MUST be cleared, if set during an erase attempt, before further attempts are allowed by the Write State Machine. SR.5 is only clear by the Clear Status Register Command, in cases where multiple blocks are erase before full status is checked. If error is detected, clear the Status Register before attempting retry or other error recovery.		

290538-5

Figure 5. Automated Block Erase Flowchart

8



Bus Operation	Command	Comments
Write	Erase Suspend	Data = B0H Addr = X
Read		Status Register Data Toggle CE# or OE# to update SRD. Addr = X
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy
Standby		Check SR.6 1 = Erase Suspended 0 = Erase Completed
Write	Read Array	Data = FFH Addr = X
Read		Read array data from block other than the one being erased.
Write	Erase Resume	Data = D0H Addr = X

290538-6

Figure 6. Erase Suspend/Resume Flowchart

3.4 Boot Block Locking

The boot block family architecture features a hardware-lockable boot block so that the kernel code for the system can be kept secure while the parameter and main blocks are programmed and erased independently as necessary. Only the boot block can be locked independently from the other blocks.

3.4.1 V_{PP} = V_{IL} FOR COMPLETE PROTECTION

For complete write protection of all blocks in the flash device, the V_{PP} programming voltage can be held low. When V_{PP} is below V_{PPLK}, any Program or Erase operation will result in a error in the Status Register.

3.4.2 WP# = V_{IL} FOR BOOT BLOCK LOCKING

When WP# = V_{IL}, the boot block is locked and any Program or Erase operation will result in an error in the Status Register. All other blocks remain unlocked in this condition and can be programmed or erased normally. Note that this feature is overridden and the boot block unlocked when RP# = V_{HH}.

3.4.3 RP# = V_{HH} OR WP# = V_{IH} FOR BOOT BLOCK UNLOCKING

Two methods can be used to unlock the boot block:

1. WP# = V_{IH}
2. RP# = V_{HH}

If both or either of these two conditions are met, the boot block will be unlocked and can be programmed or erased. The Truth Table, Table 8, clearly defines the write protection methods.

Table 8. Write Protection Truth Table for SmartVoltage Boot Block Family

V_{PP}	RP#	WP#	Write Protection Provided
V_{IL}	X	X	All Blocks Locked
≥V_{PPLK}	V_{IL}	X	All Blocks Locked (Reset)
≥V_{PPLK}	V_{HH}	X	All Blocks Unlocked
≥V_{PPLK}	V_{IH}	V_{IL}	Boot Block Locked
≥V_{PPLK}	V_{IH}	V_{IH}	All Blocks Unlocked

3.5 Power Consumption

3.5.1 ACTIVE POWER

With CE# at a logic-low level and RP# at a logic-high level, the device is placed in the active mode. Refer to the DC Characteristics table for I_{CC} current values.

3.5.2 AUTOMATIC POWER SAVINGS (APS)

Automatic Power Savings (APS) is a low-power feature during active mode of operation. The boot block flash memory family incorporates Power Reduction Control (PRC) circuitry which allows the device to put itself into a low current state when it is not being accessed. After data is read from the memory array, PRC logic controls the device's power consumption by entering the APS mode where typical I_{CC} current is less than1 mA. The device stays in this static state with outputs valid until a new location is read.

3.5.3 STANDBY POWER

With CE# at logic high level (V_{IH}), and the CUI in read mode, the memory is placed in standby mode. The Standby operation disables much of the device's circuitry and substantially reduces device power consumption. The outputs (DQ[0:15] or DQ[0:7]) are placed in a high-impedance state independent of the status of the OE# signal. When CE# is at logic-high level during erase or program functions, the devices will continue to perform the erase or program function and consume erase or program active power until erase or program is completed.

3.5.4 DEEP POWER-DOWN MODE

The SmartVoltage boot block family supports a low typical I_{CC} in deep power-down mode. The device has a RP# pin which places the device in the deep power-down mode. When RP# is at a logic-low (GND ± 0.2V), all circuits are turned off in order to save power. (Note: BYTE# pin must be at CMOS levels to achieve the most deep power-down current savings.)

During read modes, the RP# pin going low de-selects the memory and places the output drivers in a high impedance state. Recovery from the deep power-down state, requires a minimum access time of t_{PHQV}. (See the AC Characteristics table for specification numbers.)

8

During erase or program modes, RP# low will abort either Erase or Program operation. The contents of the memory are no longer valid as the data has been corrupted by the RP# function. As in the read mode above, all internal circuitry is turned off to achieve the power savings.

RP# transitions to V_{IL}, or turning power off to the device will clear the Status Register.

3.6 Power-Up Operation

The device is designed to offer protection against accidental block erasure or programming during power transitions. Upon power-up, the device is indifferent as to which power supply, V_{PP} or V_{CC}, powers-up first. Power supply sequencing is not required.

A system designer must guard against spurious writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and CE# must be low for a command write, driving either signal to V_{IH} will inhibit writes to the device. The CUI architecture provides an added level of protection since alteration of memory contents can only occur after successful completion of the two-step command sequences. Finally the device is disabled until RP# is brought to V_{IH}, regardless of the state of its control inputs. By holding the device in reset (RP# connected to system PowerGood) during power up/down, invalid bus conditions that may occur can be masked. This feature provides yet another level of memory protection.

3.6.1 RP# CONNECTED TO SYSTEM RESET

The use of RP# during system reset is important with automated write/erase devices. When the system comes out of reset it expects to read from the flash memory. Automated flash memories provide status information when accessed during writes/erase modes. If a CPU reset occurs with no flash memory reset, proper CPU initialization would not occur because the flash memory would be providing the status information instead of array data. Intel's flash memories allow proper CPU initialization following a system reset through the use of the RP# input. In this application RP# is controlled by the same RESET# signal that resets the system CPU.

3.7 Power Supply Decoupling

Flash memory's power switching characteristics require careful device decoupling methods. System designers should consider three supply current issues:

1. Standby current levels (I_{CCS})
2. Active current levels (I_{CCR})
3. Transient peaks produced by falling and rising edges of CE#.

Transient current magnitudes depend on the device outputs' capacitive and inductive loading. Two-line control and proper decoupling capacitor selection will suppress these transient voltage peaks. Each flash device should have a 0.1 μF ceramic capacitor connected between each V_{CC} and GND, and between its V_{PP} and GND. These high frequency, inherently low inductance capacitors should be placed as close as possible to the package leads.

3.7.1 V_{PP} TRACE ON PRINTED CIRCUIT BOARDS

Writing to flash memories while they reside in the target system, requires special consideration of the V_{PP} power supply trace by the printed circuit board designer. The V_{PP} pin supplies the flash memory cells current for programming and erasing. One should use similar trace widths and layout considerations given to the V_{CC} power supply trace. Adequate V_{PP} supply traces, and decoupling capacitors placed adjacent to the component, will decrease spikes and overshoots.

3.7.2 V_{CC}, V_{PP} AND RP# TRANSITIONS

The CUI latches commands as issued by system software and is not altered by V_{PP} or CE# transitions or WSM actions. Its default state upon power-up, after exit from deep power-down mode, or after V_{CC} transitions above V_{LKO} (Lockout voltage), is read array mode.

After any Word/Byte Write or Block Erase operation is complete and even after V_{PP} transitions down to V_{PPLK}, the CUI must be reset to read array mode via the Read Array command when accesses to the flash memory are desired.

4.0 ABSOLUTE MAXIMUM RATINGS*

Operating Temperature

During Read ..... −40°C to +125°C

During Block Erase and Word/Byte Write ..... −40°C to +125°C

Temperature Under Bias ..... −40°C to +125°C

Storage Temperature ..... −65°C to +125°C

Voltage on Any Pin (except V_{CC}, V_{PP}, A_9 and RP#) with Respect to GND ..... −2.0V to +7.0V(1)

Voltage on Pin RP# or Pin A_9 with Respect to GND ..... −2.0V to +13.5V(1,2)

V_{PP} Program Voltage with Respect to GND during Block Erase and Word/Byte Write ..... −2.0V to +14.0V(1,2)

V_{CC} Supply Voltage with Respect to GND ..... −2.0V to +7.0V(1)

Output Short Circuit Current ..... 100 mA (3)

NOTICE: This data sheet contains information on products in the sampling and initial production phases of development. The specifications are subject to change without notice. Verify with your local Intel Sales office that you have the latest data sheet before finalizing a design.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:

1. Minimum DC voltage is −0.5V on input/output pins. During transitions, this level may undershoot to −2.0V for periods <20 ns. Maximum DC voltage on input/output pins is V_{CC} + 0.5V which, during transitions, may overshoot to V_{CC} + 2.0V for periods <20 ns.
2. Maximum DC voltage on V_{PP} may overshoot to +14.0V for periods <20ns. Maximum DC voltage on RP# or A_9 may overshoot to 13.5V for periods <20 ns.
3. Output shorted for no more than one second. No more than one output shorted at a time.

8

5.0 OPERATING CONDITIONS

Table 9. Temperature and V_{CC} Operating Conditions

Symbol	Parameter	Notes	Min	Max	Units
T_A	Operating Temperature		−40	+125	°C
V_{CC}	V_{CC} Supply Voltage (10%)		4.50	5.50	Volts

5.1 V_{CC} Voltage

Table 10. V_{CC} Supply Tming

Symbol	Parameter	Notes	Min	Max	Unit
T_{5VPH}	V_{CC} at 4.5V (minimum) to RP# High	1	2		μs

NOTE:
1. The T_{5VPH} time must be strictly followed to guarantee all other read and write specifications.

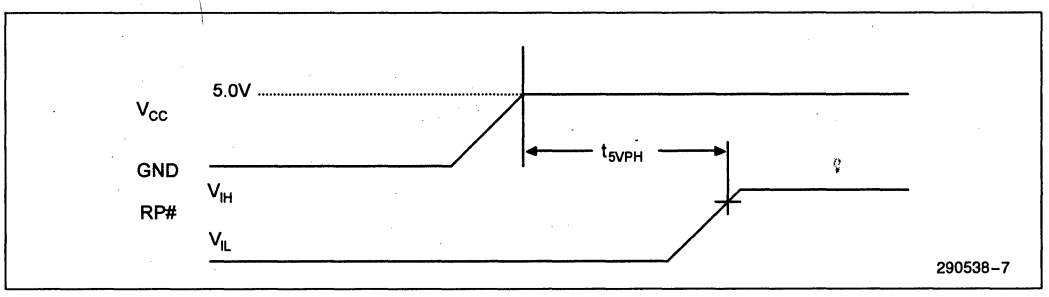


Figure 7. V_{CC} Supply Waveform

5.2 DC Characteristics

Table 11. DC Characteristics: Automotive Temperature Operation

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Conditions
I_{IL}	Input Load Current	1			± 1.0	µA	$V_{CC} = V_{CC}$MAX $V_{IN} = V_{CC}$ or GND
I_{LO}	Output Leakage Current	1			± 10	µA	$V_{CC} = V_{CC}$ MAX $V_{IN} = V_{CC}$ or GND
I_{CCS}	V_{CC} Standby Current	1,3		0.8	2.5	mA	$V_{CC} = V_{CC}$ MAX CE# = RP# = BYTE# = V_{IH}
				70	150	µA	$V_{CC} = V_{CC}$ MAX CE# = RP# = WP# = V_{CC} ± 0.2V
I_{CCD}	V_{CC} Deep Power-Down Current	1		0.2	10	µA	$V_{CC} = V_{CC}$ MAX $V_{IN} = V_{CC}$ or GND RP# = GND ± 0.2V
I_{CCR}	V_{CC} Read Current for Word or Byte	1,5,6		50	65	mA	CMOS $V_{CC} = V_{CC}$ MAX CE = V_{IL} f = 10 Mhz (5V) 5 Mhz (3.3V) I_{OUT} = 0 mA Inputs = GND ± 0.2V or V_{CC} ± 0.2V
				55	70	mA	TTL $V_{CC} = V_{CC}$ MAX CE# = V_{IL} f = 10 MHz I_{OUT} = 0 mA Inputs = V_{IL} or V_{IH}
I_{CCW}	V_{CC} Write Current for Word or Byte	1,4		25	50	mA	Word/Byte Program in Progress $V_{PP} = V_{PPH}1$ (at 5V)
				20	45	mA	Word/Byte Program in Progress $V_{PP} = V_{PPH}2$ (at 12V)

8

Table 11. DC Characteristics: Automotive Temperature Operation (Continued)

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Conditions
I_{CCE}	V_{CC} Erase Current	1,4		22	45	mA	Block Erase in Progress $V_{PP} = V_{PPH}1$ (at 5V)
				18	40	mA	Block Erase in Progress $V_{PP} = V_{PPH}2$ (at 12V)
I_{CCES}	V_{CC} Erase Suspend Current	1,2		5	12.0	mA	CE# = V_{IH} Block Erase Suspend $V_{PP} = V_{PPH}1$ (at 5V)
I_{PPS}	V_{PP} Standby Current	1		± 5	± 15	µA	$V_{PP} \leq V_{CC}$
I_{PPD}	V_{PP} Deep Power-down Current	1		0.2	10	µA	RP# = GND ± 0.2V
I_{PPR}	V_{PP} Read Current	1		50	200	µA	$V_{PP} > V_{CC}$
I_{PPW}	V_{PP} Write Current for Word/Byte	1		13	30	mA	$V_{PP} = V_{PPH}$ Word Write in Progress $V_{PP} = V_{PPH}1$ (at 5V)
				8	25	mA	$V_{PP} = V_{PPH}$ Word Write in Progress $V_{PP} = V_{PPH}2$ (at 12V)
I_{PPE}	V_{PP} Erase Current	1		15	25	mA	$V_{PP} = V_{PPH}$ Block Erase in Progress $V_{PP} = V_{PPH}1$ (at 5V)
				10	20	mA	$V_{PP} = V_{PPH}$ Block Erase in Progress $V_{PP} = V_{PPH}2$ (at 12V)
I_{PPES}	V_{PP} Erase Suspend Current	1		50	200	µA	$V_{PP} = V_{PPH}$ Block Erase Suspend in Progress
$I_{RP\#}$	RP# Boot Block Unlock Current	1,4			500	µA	RP# = V_{HH} V_{PP} = 12V
I_{ID}	A_9 Intelligent Identifier Current	1,4			500	µA	$A_9 = V_{ID}$
V_{ID}	A_9 Intelligent Identifier Voltage		11.4		12.6	V	
V_{IL}	Input Low Voltage		−0.5		0.8	V	

Table 11. DC Characteristics: Automotive Temperature Operation (Continued)

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Conditions
V_{IH}	Input High Voltage		2.0		$V_{CC} \pm 0.5V$	V	
V_{OL}	Output Low Voltage (TTL)				0.45	V	$V_{CC} = V_{CC}MIN$ $I_{OL} = 5.8$ mA $V_{PP} = 12V$
$V_{OH}1$	Output High Voltage (TTL)		2.4			V	$V_{CC} = V_{CC}$ MIN $I_{OH} = -1.5$ mA
$V_{OH}2$	Output High Voltage (CMOS)		$V_{CC} - .4V$			V	$V_{CC} = V_{CC}$ MIN $I_{OH} = -100\ \mu A$
V_{PPLK}	V_{PP} Lock-Out Voltage	3	0.0		1.5	V	Complete Write Protection
$V_{PPH}1$	V_{PP} (Program/Erase Operations)	7,8	4.5		5.5	V	V_{PP} at 5V
$V_{PPH}2$	V_{PP} (Program/Erase Operations)	7,8	11.4		12.6	V	V_{PP} at 12V
V_{LKO}	V_{CC} Erase/Write Lock Voltage	10	3.9			V	$V_{PP} = 12V$
V_{HH}	RP# Unlock Voltage		11.4		12.6	V	Boot Block Write/Erase $V_{PP} = 12V$

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at $V_{CC} = 5.0V$, $T = 25°C$. These currents are valid for all product versions (packages and speeds).
2. I_{CCES} is specified with the device de-selected. If the devices is read while in erase suspend mode, current draw is the sum of I_{CCES} and I_{CCR}.
3. Block erases and word/byte writes are inhibited when $V_{PP} = V_{PPLK}$, and not guaranteed in the range between $V_{PPH}1$ and V_{PPLK}.
4. Sampled, not 100% tested.
5. Automatic Power Savings (APS) reduces I_{CCR} to less than 1 mA typical, in static operation.
6. CMOS Inputs are either $V_{CC} \pm 0.2V$ or GND $\pm$ 0.2V. TTL Inputs are either V_{IL} or V_{IH}.

8

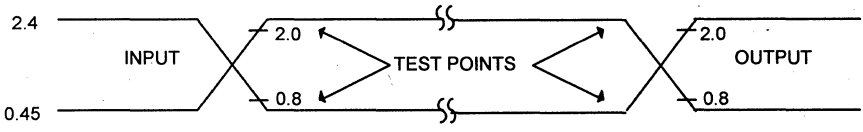
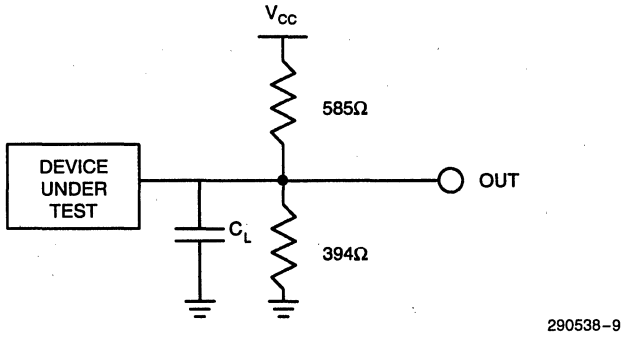


Figure 8. 5V Inputs and Measurement Points



NOTE:
C_L = 100 pF, includes Jig Capacitance

Figure 9. 5V Standard Test Configuration

5.3 AC Characteristics

Table 12. AC Characteristics: Read Only Operations[1] (Automotive Temperature)

Symbol	Parameter	Note	Min	Max	Unit
t_{AVAV}	Read Cycle Time		80		ns
t_{AVQV}	Address to Output Delay			80	ns
t_{ELQV}	CE# to Output Delay	2		80	ns
t_{PHQV}	RP# to Output Delay			450	ns
t_{GLQV}	OE# to Output Delay	2		40	ns
t_{ELQX}	CE# to Output in Low Z	3	0		ns
t_{EHQZ}	CE# to Output in High Z	3		30	ns
t_{GLQX}	OE# to Output in Low Z	3	0		ns
t_{GHQZ}	OE# to Output in High Z	3		30	ns
t_{OH}	Output Hold from Address CE#, or OE# Change Whichever Occurs First	3	0	0	ns
t_{ELFL} t_{ELFH}	CE# Low to BYTE High or Low	3		5	ns
t_{AVFL}	Address to BYTE# High or Low	3		5	ns
t_{FLQV} t_{FHQV}	BYTE# to Output Delay	3,4		80	ns
t_{FLQZ}	BYTE# low to Output in High Z	3		30	ns

NOTES:
1. See AC Input/Output Reference Waveform for timing measurements.
2. OE# may be delayed up to $t_{CE}-t_{OE}$ after the falling edge of CE# without impact on t_{CE}.
3. Sampled, but not 100% tested.
4. t_{FLQV}, BYTE# switching low to valid output delay will be equal to t_{AVQV}, measured from the time DQ_{15}/A_{-1} becomes valid.
5. See 5V Standard Test Configuration. (Figure 9)

8

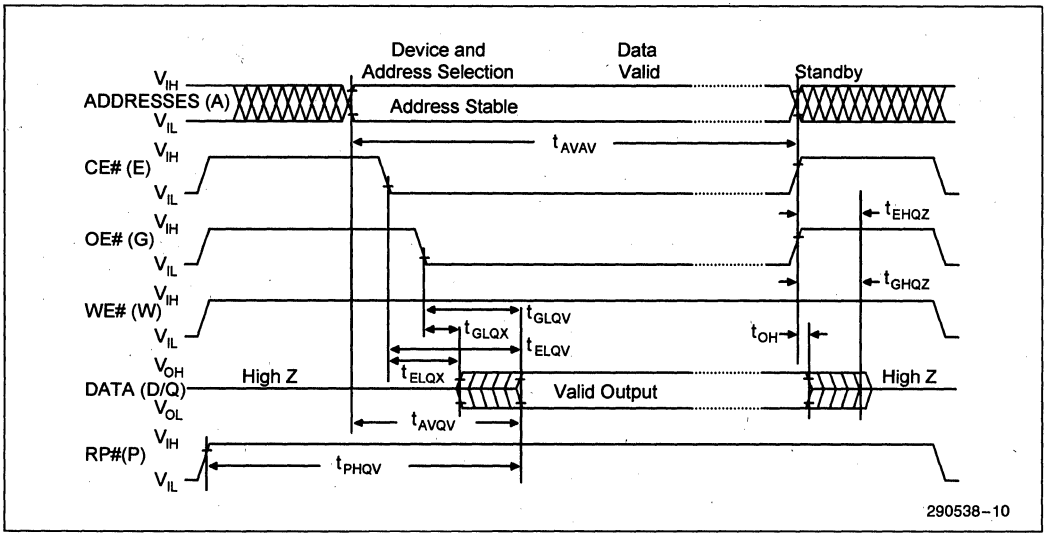


Figure 10. AC Waveforms for Read Operations

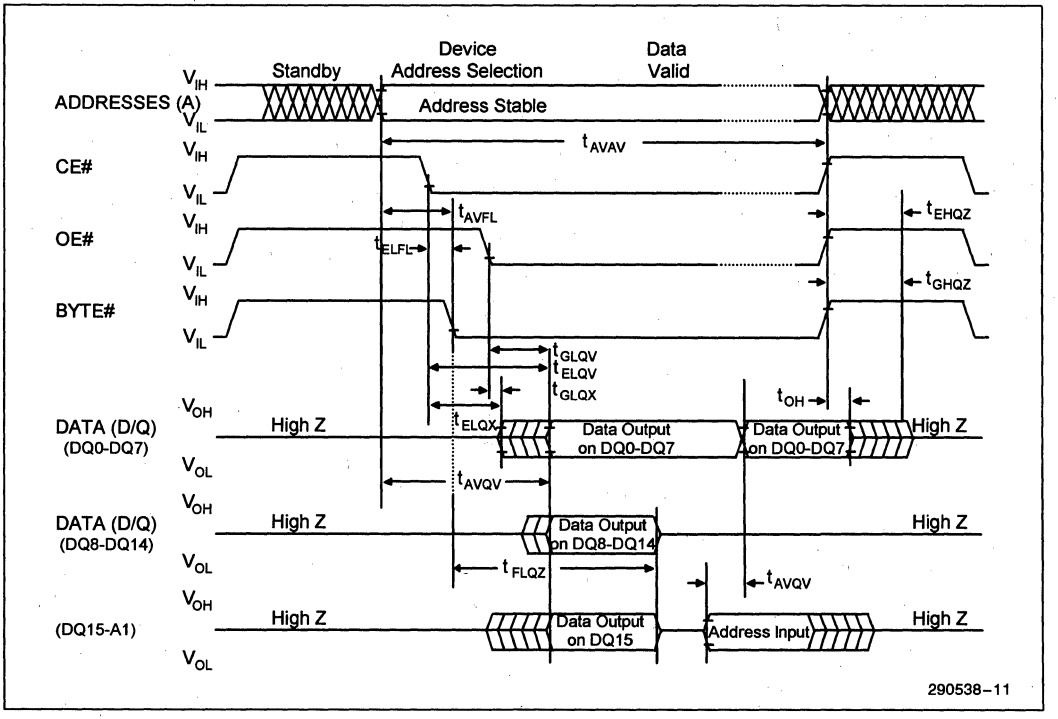


Figure 11. BYTE# Timing Diagram for Both Read and Write Operations with V_{CC} at 5V

Table 13. AC Characteristics: WE#—Controlled Write Operations[1] (Automotive Temperature)

Symbol	Parameter	Notes	Min	Max	Unit
t_{AVAV}	Write Cycle Time		80		ns
t_{PHWL}	RP# High Recovery to WE# Going Low		220		ns
t_{ELWL}	CE# Setup to WE# Going Low		0		ns
t_{PHHWH}	Boot Block Lock Setup to WE# Going High	6,8	100		ns
t_{VPWH}	V_{PP} Setup to WE# Going High	5,8	100		ns
t_{AVWH}	Address Setup to WE# Going High	3	60		ns
t_{DVWH}	Data Setup to WE# Going High	4	60		ns
t_{WLWH}	WE# Pulse Width		60		ns
t_{WHDX}	Data Hold Time from WE# High	4	0		ns
t_{WHAX}	Address Hold Time from WE# High	3	10		ns
t_{WHEH}	CE# Hold Time from WE# High		10		ns
t_{WHWL}	WE# Pulse Width High		20		ns
t_{WHQV1}	Duration of Word/Byte Write Operation	2,5	7		μs
t_{WHQV2}	Duration of Erase Operation (Boot)	2,5,6	0.4		s
t_{WHQV3}	Duration of Erase Operation (Parameter)	2,5	0.4		s
t_{WHQV4}	Duration of Erase Operation (Main)	2,5	0.7		s
t_{QWL}	V_{PP} Hold from Valid SRD	5,8	0		ns
t_{QVPH}	RP# V_{HH} Hold from Valid SRD	6,8	0		ns
t_{PHBR}	Boot-Block Relock Delay	7,8		100	ns

NOTES:
1. Read timing characteristics during write and erase operations are the same as during read-only operations. Refer to AC Characteristics during read mode.
2. The on-chip WSM completely automates Program/Erase operations; Program/Erase algorithms are now controlled internally which includes verify and margining operations.
3. Refer to command definition table for valid A_{IN}.
4. Refer to command definition table for valid D_{IN}.
5. Program/Erase durations are measured to valid SRD data (successful operation, SR.7 = 1)
6. For boot block Program/Erase, RP# should be held at V_{HH} or WP# should be held at V_{IH} until operation completes successfully.
7. Time t_{PHBR} is required for successful relocking of the boot block.
8. Sampled, but not 100% tested.
9. V_{PP} at 5.0V.
10. V_{PP} at 12.0V.
11. See 5V Standard Test Configuration.

8

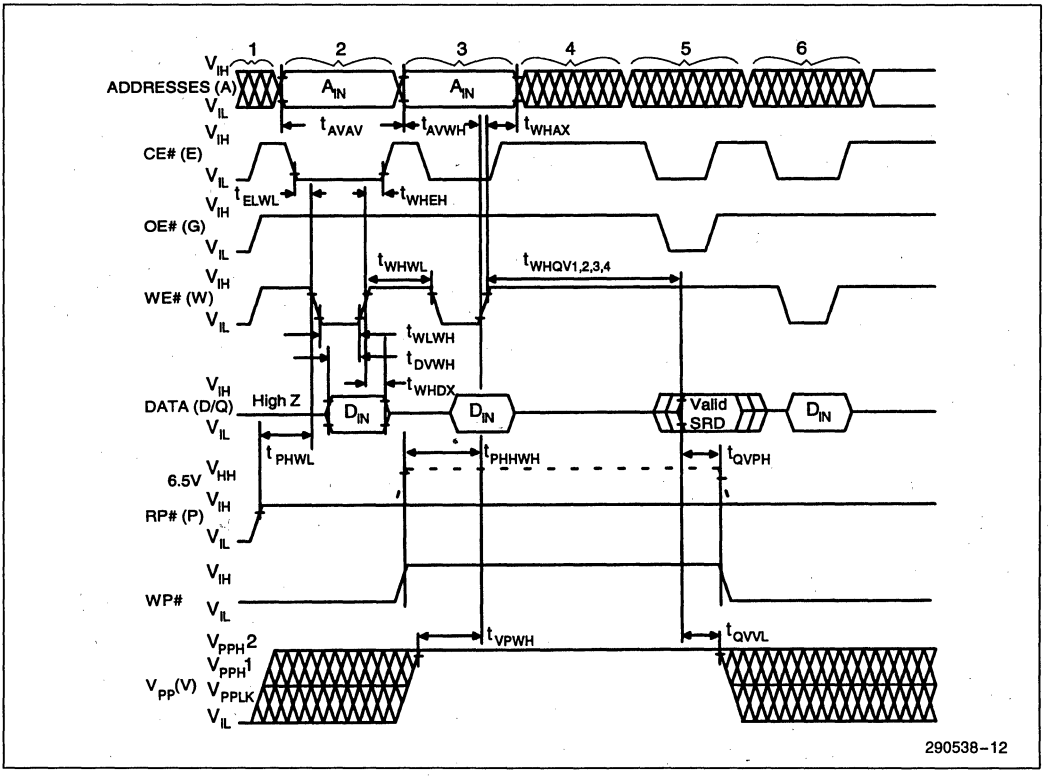


Figure 12. AC Waveforms for Write and Erase Operations (WE#-Controlled Writes)

Table 14. AC Characteristics: CE#—Controlled Write Operations (1,12)

Symbol	Parameter	Notes	Min	Max	Unit
t_{AVAV}	Write Cycle Time		80		ns
t_{PHEL}	RP# High Recovery to CE# Going Low			300	ns
t_{WLEL}	WE# Setup to CE# Going Low		0		ns
t_{PHHEH}	Boot Block Lock Setup to CE# Going High	6,8	100		ns
t_{VPEH}	V_{PP} Setup to CE# Going High	5,8	100		ns
t_{AVEH}	Address Setup to CE# Going High		60		ns
t_{DVEH}	Data Setup to CE# Going High	3	60		ns
t_{ELEH}	CE# Pulse Width	4	60		ns
t_{EHDX}	Data Hold Time from CE# High		0		ns
t_{EHAX}	Address Hold Time from CE# High	4	10		ns
t_{EHWH}	WE# Hold Time from CE# High	3	10		ns
t_{EHEL}	CE# Pulse Width High		20		ns
t_{EHQV1}	Duration of Word/Byte Write Operation	2,5	7		μs
t_{EHQV2}	Duration of Erase Operation (Boot)	2,5,6	0.4		s
t_{EHQV3}	Duration of Erase Operation (Parameter)	2,5	0.4		s
t_{EHQV4}	Duration of Erase Operation (Main)	2,5	0.7		s
t_{QWL}	V_{PP} Hold from Valid SRD	5,8	0		ns
t_{QVPH}	RP# V_{HH} Hold from Valid SRD	6,8	0		ns
t_{PHBR}	Boot-Block Relock Delay	7,8		100	ns

NOTES:
See WE# Controlled Write Operations for notes 1 through 11.
12. Chip-Enable controlled writes: Write operations are driven by the valid combination of CE# and WE# in systems where CE# defines the write pulse-width (within a longer WE# timing waveform), all set-up, hold and inactive WE# times should be measured relative to the CE# waveform.

8

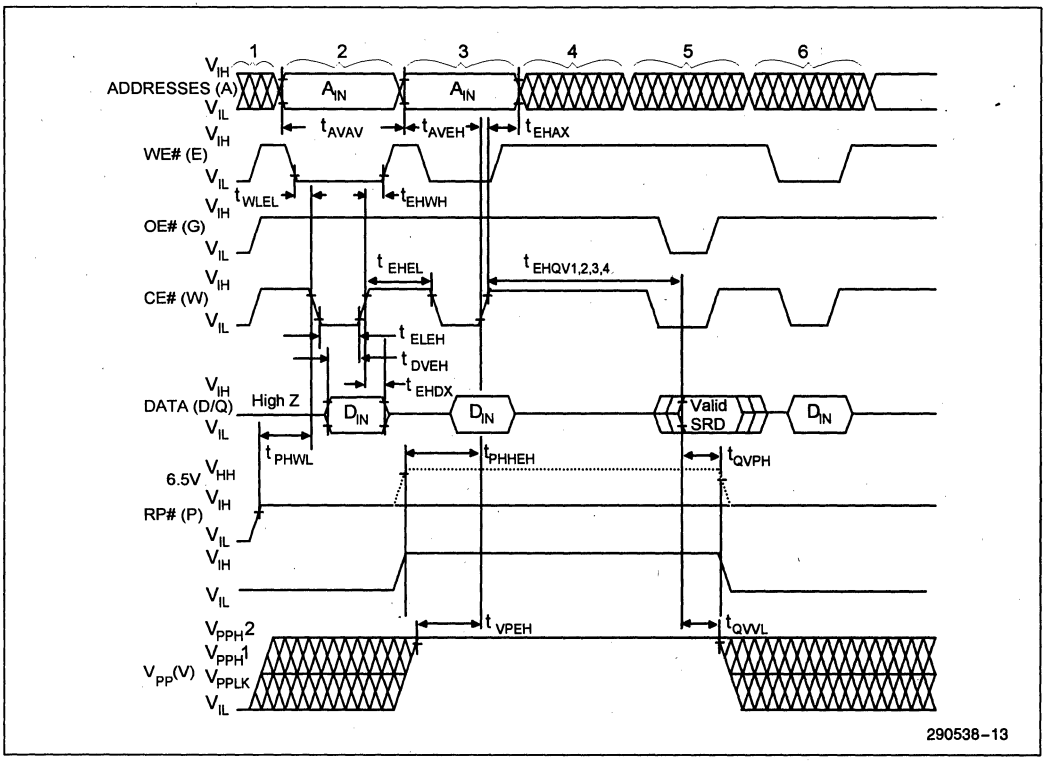


Figure 13. Alternate AC Waveforms for Write and Erase Operations (CE#-Controlled Writes)

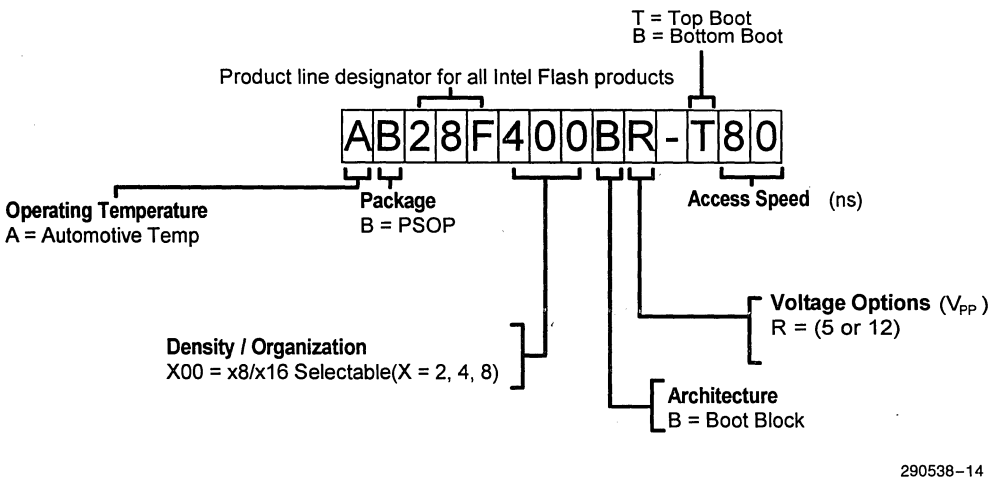
Table 15. Extended Temperature Operations—Erase and Program Timings

Parameter	$V_{PP} = 5V \pm 10\%$		$V_{PP} = 12V \pm 5\%$		Unit
	Typ	Max	Typ	Max	
Boot/Parameter Block Erase Time	0.8	7.8	0.34	4.0	s
Main Block Erase Time	1.9	15.4	1.1	7.1	s
Main Block Write Time (Byte Mode)	1.4	16.8	1.2	6.8	s
Main Block Write Time (Word Mode)	0.9	8.4	0.6	3.4	s

All numbers are sampled, not 100% tested.

6.0 ADDITIONAL INFORMATION

6.1 Ordering Information



VALID COMBINATIONS:
AB28F400BR-T80 AB28F400BR-B80

6.2 References

Order Number	Document
292130	AB-57 "Boot Block Architecture for Safe Firmware Updates"
292154	AB-60 "2/4/8-Mbit SmartVoltage Boot Block Flash Memory Family"
292098	AP-363 "Extended Flash BIOS Concepts for Portable Computers"
292148	AP-604 "Using Intel's Boot Block Flash Memory Parameter Blocks to Replace EEPROM"
290448	28F002/200BX-T/B 2-Mbit Boot Block Flash Memory Datasheet
290449	28F002/200BL-T/B 2-Mbit Low Power Boot Block Flash Memory Datasheet
290450	28F004/400BL-T/B 4-Mbit Low Power Boot Block Flash Memory Datasheet
290451	28F004/400BX-T/B 4-Mbit Boot Block Flash Memory Datasheet
290531	2-Mbit SmartVoltage Boot Block Flash Memory Family Datasheet
290539	8-Mbit SmartVoltage Boot Block Flash Memory Family Datasheet

6.3 Revision History

Number	Description
-001	Initial release of datasheet

A28F400BX-T/B
4-MBIT (256K x16, 512K x8) BOOT BLOCK FLASH MEMORY FAMILY

Automotive

- x8/x16 Input/Output Architecture
  - A28F400BX-T, A28F400BX-B
  - For High Performance and High Integration 16-bit and 32-bit CPUs
- Optimized High Density Blocked Architecture
  - One 16 KB Protected Boot Block
  - Two 8 KB Parameter Blocks
  - One 96 KB Main Block
  - Three 128 KB Main Blocks
  - Top or Bottom Boot Locations
- Extended Cycling Capability
  - 1,000 Block Erase Cycles
- Automated Word/Byte Write and Block Erase
  - Command User Interface
  - Status Register
  - Erase Suspend Capability
- SRAM-Compatible Write Interface
- Automatic Power Savings Feature
  - 1 mA Typical I_{CC} Active Current in Static Operation
- Very High-Performance Read
  - 90 ns Maximum Access Time
  - 45 ns Maximum Output Enable Time
- Low Power Consumption
  - 25 mA Typical Active Read Current
- Deep Power-Down/Reset Input
  - Acts as Reset for Boot Operations
- Automotive Temperature Operation
  - $-40°C$ to $+125°C$
- Write Protection for Boot Block
- Hardware Data Protection Feature
  - Erase/Write Lockout During Power Transitions
- Industry Standard Surface Mount Packaging
  - JEDEC ROM Compatible 44-Lead PSOP
- 12V Word/Byte Write and Block Erase
  - $V_{PP} = 12V \pm 5\%$ Standard
- ETOX™ III Flash Technology
  - 5V Read

8

November 1994
Order Number: 290501-002

Intel's 4-Mbit Flash Memory Family is an extension of the Boot Block Architecture which includes block-selective erasure, automated write and erase operations and standard microprocessor interface. The 4-Mbit Flash Memory Family enhances the Boot Block Architecture by adding more density and blocks, x8/x16 input/output control, very high speed, low power, an industry standard ROM compatible pinout and surface mount packaging. The 4-Mbit flash family is an easy upgrade from Intel's 2-Mbit Boot Block Flash Memory Family.

The Intel A28F400BX-T/B are 16-bit wide flash memory offerings optimized to meet the rigorous environmental requirements of Automotive Applications. These high density flash memories provide user selectable bus operation for either 8-bit or 16-bit applications. The A28F400BX-T and A28F400BX-B are 4,194,304-bit non-volatile memories organized as either 524,288 bytes or 262,144 words of information. They are offered in 44-Lead plastic SOP packages. The x8/x16 pinout conforms to the industry standard ROM/EPROM pinout. Read and Write characteristics are guaranteed over the ambient temperature range of $-40°C$ to $+125°C$.

These devices use an integrated Command User Interface (CUI) and Write State Machine (WSM) for simplified word/byte write and block erasure. The A28F400BX-T provide block locations compatible with Intel's MCS-186 family, 80286, i386™, i486™, i860™ and 80960CA microprocessors. The A28F400BX-B provides compatibility with Intel's 80960KX and 80960SX families as well as other embedded microprocessors.

The boot block includes a data protection feature to protect the boot code in critical applications. With a maximum access time of 90 ns, these 4-Mbit flash devices are very high performance memories which interface at zero-wait-state to a wide range of microprocessors and microcontrollers.

Manufactured on Intel's 0.8 micron ETOX™ III process, the 4-Mbit flash memory family provides world class quality, reliability and cost-effectiveness at the 4-Mbit density level.

1.0 PRODUCT FAMILY OVERVIEW

Throughout this datasheet the A28F400BX refers to both the A28F400BX-T and A28F400BX-B devices. Section 1 provides an overview of the 4-Mbit flash memory family including applications, pinouts and pin descriptions. Section 2 describes in detail the specific memory organization for the A28F400BX. Section 3 provides a description of the family's principles of operations. Finally the family's operating specifications are described.

1.1 Main Features

The A28F400BX boot block flash memory family is a very high performance 4-Mbit (4,194,304 bit) memory family organized as either 256-KWords (262,144 words) of 16 bits each or 512-Kbytes (524,288 bytes) of 8 bits each.

Seven Separately Erasable Blocks including a **Hardware-Lockable boot block** (16,384 Bytes), **Two parameter blocks** (8,192 Bytes each) and **Four main blocks** (1 block of 98,304 Bytes and 3 blocks of 131,072 Bytes) are included on the 4-Mbit family. An erase operation erases one of the main blocks in typically 3 seconds and the boot or parameter blocks in typically 1.5 seconds independent of the remaining blocks. Each block can be independently erased and programmed 1,000 times.

The Boot Block is located at either the top (A28F400BX-T) or the bottom (A28F400BX-B) of the address map in order to accommodate different microprocessor protocols for boot code location. The **hardware lockable boot block** provides the most secure code storage. The boot block is intended to store the kernel code required for booting-up a system. When the **RP#** pin is between 11.4V and 12.6V the boot block is unlocked and program and erase operations can be performed. When the RP# pin is at or below 6.5V the boot block is locked and program and erase operations to the boot block are ignored.

The A28F400BX products are available in the ROM/EPROM compatible pinout and housed in the 44-Lead PSOP (Plastic Small Outline) package as shown in Figure 3.

The **Command User Interface (CUI)** serves as the interface between the microprocessor or microcontroller and the internal operation of the A28F400BX flash memory.

Program and Erase Automation allows program and erase operations to be executed using a two-write command sequence to the CUI. The internal Write State Machine (WSM) automatically executes the algorithms and timings necessary for program and erase operations, including verifications, thereby unburdening the microprocessor or microcontroller. Writing of memory data is performed in word or byte increments typically within 9 μs which is a 100% improvement over previous flash memory products.

The **Status Register (SR)** indicates the status of the WSM and whether the WSM successfully completed the desired program or erase operation.

Maximum Access Time of **90 ns (TACC)** is achieved over the automotive temperature range, 10% V_{CC} supply range (4.5V to 5.5V) and 100 pF output load.

I_{PP} maximum Program current is 40 mA for x16 operation and 30 mA for x8 operation. I_{PP} Erase current is 30 mA maximum. V_{PP} erase and programming voltage is 11.4V to 12.6V (V_{PP} = 12V ± 5%) under all operating conditions. Typical I_{CC} Active Current of 25 mA is achieved.

The 4-Mbit boot block flash memory family is also designed with an Automatic Power Savings (APS) feature to minimize system battery current drain and allows for very low power designs. Once the device is accessed to read array data, APS mode will immediately put the memory in static mode of operation where I_{CC} active current is typically 1 mA until the next read is initiated.

When the CE# and RP# pins are at V_{CC} and the BYTE# pin is at either V_{CC} or GND the **CMOS Standby** mode is enabled where **I_{CC}** is typically **80 μA.**

A **Deep Power-Down Mode** is enabled when the RP# pin is at ground minimizing power consumption and providing write protection during power-up conditions. **I_{CC} current** during deep power-down mode is **50 μA typical.** An initial maximum access time or Reset Time of 300 ns is required from RP# switching until outputs are valid. Equivalently, the device has a maximum wake-up time of 210 ns until writes to the Command User Interface are recognized. When RP# is at ground the WSM is reset, the Status Register is cleared and the entire device is protected from being written to. This feature prevents data corruption and protects the code stored in the device during system reset. The system Reset pin can be tied to RP# to reset the memory to normal read mode upon activation of the Reset pin. With on-chip program/erase automation in the 4-Mbit family and the RP# functionality for data protection, when the CPU is reset and even if a program or erase command is issued, the device will not recognize any operation until RP# returns to its normal state.

For the A28F400BX, Byte-wide or Word-wide Input/Output Control is possible by controlling the BYTE# pin. When the BYTE# pin is at a logic low the device is in the byte-wide mode (x8) and data is read and written through DQ[0:7]. During the byte-wide mode, DQ[8:14] are tri-stated and DQ15/A-1 becomes the lowest order address pin. When the BYTE# pin is at a logic high the device is in the word-wide mode (x16) and data is read and written through DQ[0:15].

1.2 Applications

The 4-Mbit boot block flash memory family combines high density, high performance, cost-effective flash memories with blocking and hardware protection capabilities. Its flexibility and versatility will reduce costs throughout the product life cycle. Flash memory is ideal for Just-In-Time production flow, reducing system inventory and costs, and eliminating component handling during the production phase.

During the product life cycle, when code updates or feature enhancements become necessary, flash memory will reduce the update costs by allowing either a user-performed code change via floppy disk or a remote code change via a serial link. The 4-Mbit boot block flash memory family provides full function, blocked flash memories suitable for a wide range of automotive applications.

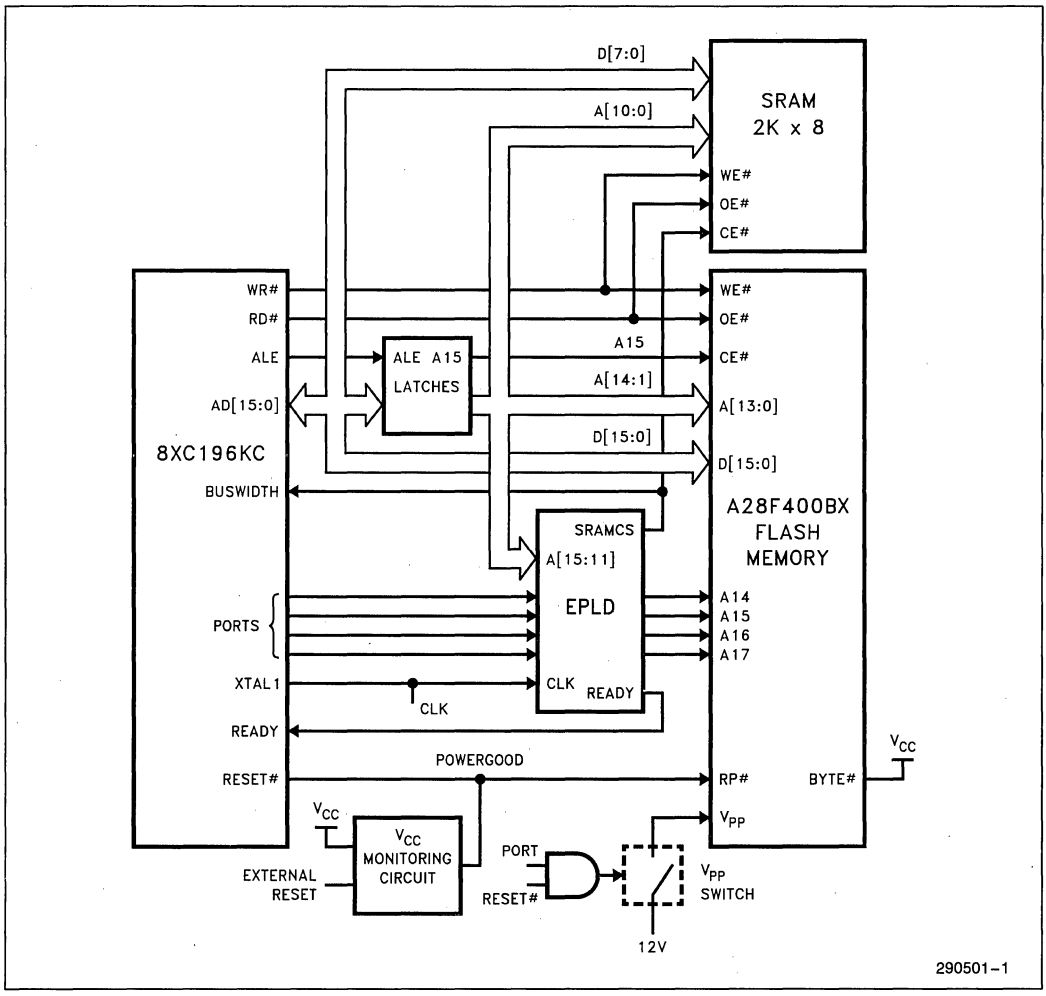


Figure 1. A28F400BX Interface to 8XC196KC

8

1.3 Pinouts

The A28F400BX 44-Lead PSOP pinout follows the industry standard ROM/EPROM pinout as shown in Figure 2.

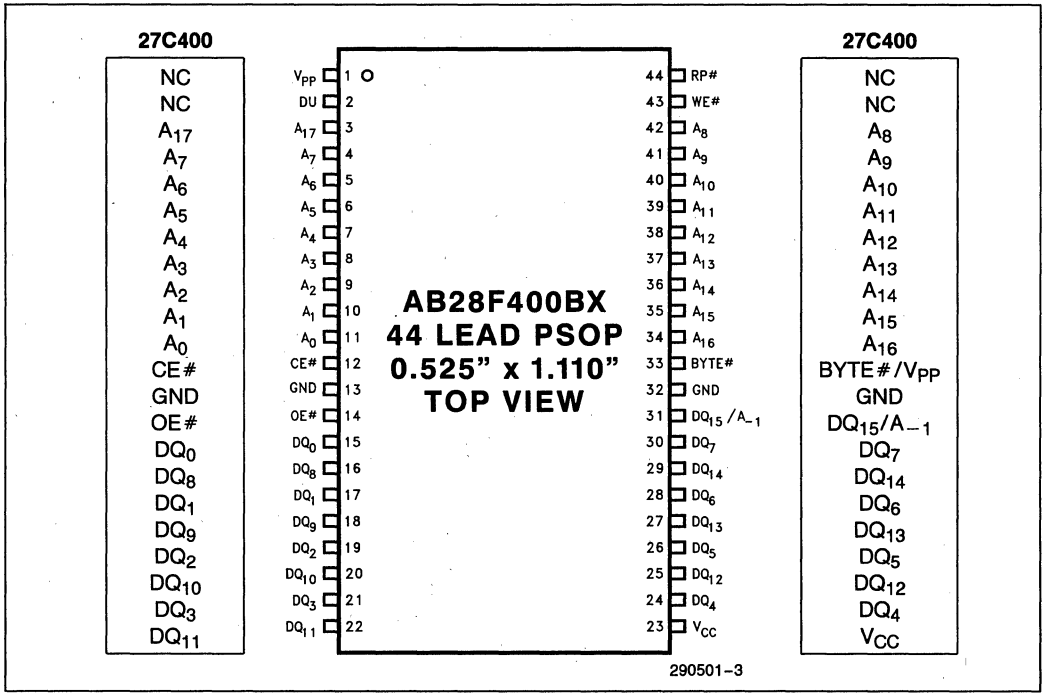


Figure 2. PSOP Lead Configuration

1.4 A28F400BX Pin Descriptions

Symbol	Type	Name and Function
A_0–A_{17}	I	**ADDRESS INPUTS** for memory addresses. Addresses are internally latched during a write cycle.
A_9	I	**ADDRESS INPUT:** When A_9 is at 12V the signature mode is accessed. During this mode A_0 decodes between the manufacturer and device ID's. When BYTE# is at a logic low only the lower byte of the signatures are read. DQ_{15}/A_{-1} is a don't care in the signature mode when BYTE# is low.
DQ_0–DQ_7	I/O	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a program command. Inputs commands to the command user interface when CE# and WE# are active. Data is internally latched during the write and program cycles. Outputs array, intelligent identifier and Status Register data. The data pins float to tri-state when the chip is deselected or the outputs are disabled.
DQ_8–DQ_{15}	I/O	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a program command. Data is internally latched during the write and program cycles. Outputs array data. The data pins float to tri-state when the chip is deselected or the outputs are disabled as in the byte-wide mode (BYTE# = "0"). In the byte-wide mode DQ_{15}/A_{-1} becomes the lowest order address for data output on DQ_0-DQ_7.
CE#	I	**CHIP ENABLE:** Activates the device's control logic, input buffers, decoders and sense amplifiers. CE# is active low; CE# high deselects the memory device and reduces power consumption to standby levels. If CE# and RP# are high, but not at a CMOS high level, the standby current will increase due to current flow through the CE# and RP# input stages.
RP#	I	**RESET/POWER-DOWN:** Provides three-state control. Puts the device in deep power-down mode. Locks the boot block from program/erase. When RP# is at logic high level and equals 6.5V maximum the boot block is locked and cannot be programmed or erased. When RP# = 11.4V minimum the boot block is unlocked and can be programmed or erased. When RP# is at a logic low level the boot block is locked, the deep power-down mode is enabled and the WSM is reset preventing any blocks from being programmed or erased, therefore providing data protection during power transitions. When RP# transitions from logic low to logic high the flash memory enters the read array mode.
OE#	I	**OUTPUT ENABLE:** Gates the device's outputs through the data buffers during a read cycle. OE# is active low.
WE#	I	**WRITE ENABLE:** Controls writes to the Command Register and array blocks. WE# is active low. Addresses and data are latched on the rising edge of the WE# pulse.
BYTE#	I	**BYTE# ENABLE:** Controls whether the device operates in the byte-wide mode (x8) or the word-wide mode (x16). BYTE# pin must be controlled at CMOS levels to meet 130 µA CMOS current in the standby mode. BYTE# = "0" enables the byte-wide mode, where data is read and programmed on DQ_0–DQ_7 and DQ_{15}/A_{-1} becomes the lowest order address that decodes between the upper and lower byte. DQ_8–DQ_{14} are tri-stated during the byte-wide mode. BYTE# = "1" enables the word-wide mode where data is read and programmed on DQ_0–DQ_{15}.
V_{PP}		**PROGRAM/ERASE POWER SUPPLY:** For erasing memory array blocks or programming data in each block. **Note:** $V_{PP} < V_{PPLMAX}$ memory contents cannot be altered.
V_{CC}		**DEVICE POWER SUPPLY (5V ± 10%)**
GND		**GROUND:** For all internal circuitry.
NC		**NO CONNECT:** Pin may be driven or left floating.
DU		**DON'T USE PIN:** Pin should not be connected to anything.

8

2.0 A28F400BX WORD/BYTE-WIDE PRODUCTS DESCRIPTION

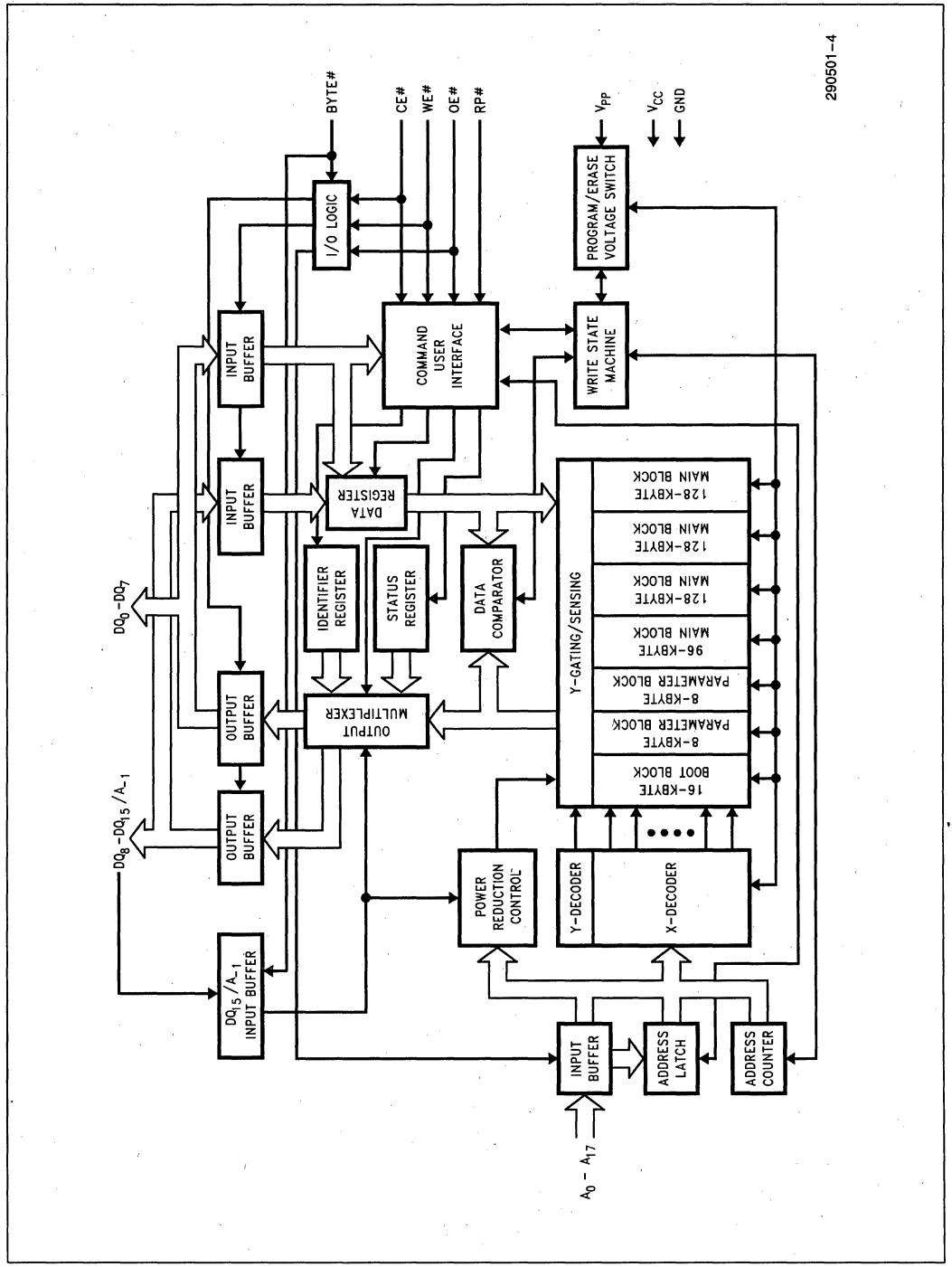


Figure 3. A28F400BX Word/Byte Block Diagram

2.1 A28F400BX Memory Organization

2.1.1 BLOCKING

The A28F400BX uses a blocked array architecture to provide independent erasure of memory blocks. A block is erased independently of other blocks in the array when an address is given within the block address range and the Erase Setup and Erase Confirm commands are written to the CUI. The A28F400BX is a random read/write memory, only erasure is performed by block.

2.1.1.1 Boot Block Operation and Data Protection

The 16-Kbyte boot block provides a lock feature for secure code storage. The intent of the boot block is to provide a secure storage area for the kernel code that is required to boot a system in the event of power failure or other disruption during code update. This lock feature ensures absolute data integrity by preventing the boot block from being written or erased when RP# is not at 12V. The boot block can be erased and written when RP# is held at 12V for the duration of the erase or program operation. This allows customers to change the boot code when necessary while providing security when needed. See the Block Memory Map section for address locations of the boot block for the A28F400BX-T and A28F400BX-B.

2.1.1.2 Parameter Block Operation

The A28F400BX has 2 parameter blocks (8 Kbytes each). The parameter blocks are intended to provide storage for frequently updated system parameters and configuration or diagnostic information. The parameter blocks can also be used to store additional boot or main code. The parameter blocks however, do not have the hardware write protection feature that the boot block has. The parameter blocks provide for more efficient memory utilization when dealing with parameter changes versus regularly blocked devices. See the Block Memory Map section for address locations of the parameter blocks for the A28F400BX-T and A28F400BX-B.

2.1.1.3 Main Block Operation

Four main blocks of memory exist on the A28F400BX (3 x 128 Kbyte blocks and 1 x 96-Kbyte blocks). See the following section on Block Memory Map for the address location of these blocks for the A28F400BX-T and A28F400BX-B products.

2.1.2 BLOCK MEMORY MAP

Two versions of the A28F400BX product exist to support two different memory maps of the array blocks in order to accommodate different microprocessor protocols for boot code location. The A28F400BX-T memory map is inverted from the A28F400BX-B memory map.

2.1.2.1. A28F400BX-B Memory Map

The A28F400BX-B device has the 16-Kbyte boot block located from 00000H to 01FFFH to accommodate those microprocessors that boot from the bottom of the address map at 00000H. In the A28F400BX-B the first 8-Kbyte parameter block resides in memory space from 02000H to 02FFFH. The second 8-Kbyte parameter block resides in memory space from 03000H to 03FFFH. The 96-Kbyte main block resides in memory space from 04000H to 0FFFFH. The three 128-Kbyte main block resides in memory space from 10000H to 1FFFFH, 20000H to 2FFFFH and 30000H to 3FFFFH (word locations). See Figure 4.

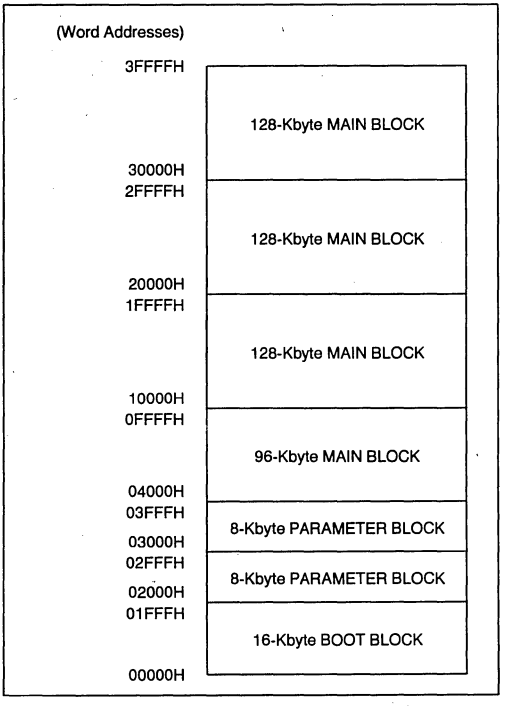


Figure 4. A28F400BX-B Memory Map

8

2.1.2.2 A28F400BX-T Memory Map

The A28F400BX-T device has the 16-Kbyte boot block located from 3E000H to 3FFFFH to accommodate those microprocessors that boot from the top of the address map. In the A28F400BX-T the first 8-Kbyte parameter block resides in memory space from 3D000H to 3DFFFH. The second 8-Kbyte parameter block resides in memory space from 3C000H to 3CFFFH. The 96-Kbyte main block resides in memory space from 30000H to 3BFFFH. The three 128-Kbyte main blocks reside in memory space from 20000H to 2FFFFH, 10000H to 1FFFFH and 00000H to 0FFFFH as shown below in Figure 5.

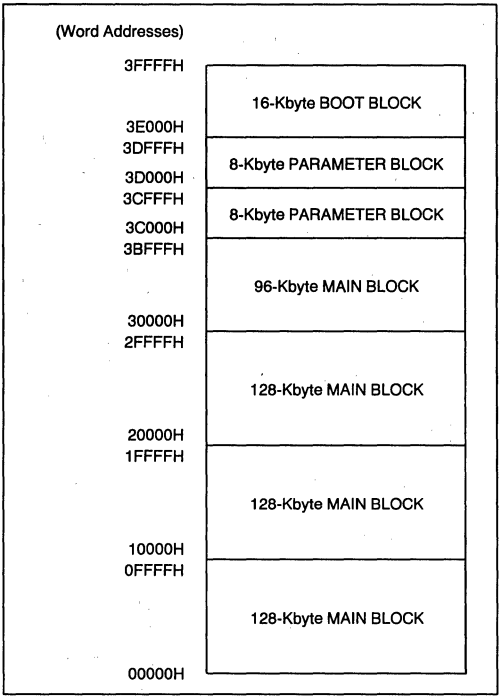


Figure 5. A28F400BX-T Memory Map

3.0 PRODUCT FAMILY PRINCIPLES OF OPERATION

Flash memory augments EPROM functionality with in-circuit electrical write and erase. The 4-Mbit flash family utilizes a Command User Interface (CUI) and internally generated and timed algorithms to simplify write and erase operations.

The CUI allows for 100% TTL-level control inputs, fixed power supplies during erasure and programming, and maximum EPROM compatibility.

In the absence of high voltage on the V_{PP} pin, the 4-Mbit boot block flash family will only successfully execute the following commands: Read Array, Read Status Register, Clear Status Register and Intelligent Identifier mode. The device provides standard EPROM read, standby and output disable operations. Manufacturer Identification and Device Identification data can be accessed through the CUI or through the standard EPROM A_9 high voltage access (V_{ID}) for PROM programming equipment.

The same EPROM read, standby and output disable functions are available when high voltage is applied to the V_{PP} pin. In addition, high voltage on V_{PP} allows write and erase of the device. All functions associated with altering memory contents: write and erase, Intelligent Identifier read and Read Status are accessed via the CUI.

The purpose of the Write State Machine (WSM) is to completely automate the write and erasure of the device. The WSM will begin operation upon receipt of a signal from the CUI and will report status back through a Status Register. The CUI will handle the WE# interface to the data and address latches, as well as system software requests for status while the WSM is in operation.

3.1 Bus Operations

Flash memory reads, erases and writes in-system via the local CPU. All bus cycles to or from the flash memory conform to standard microprocessor bus cycles.

Table 1. Bus Operations for WORD-WIDE Mode (BYTE# = V_{IH})

Mode	Notes	RP#	CE#	OE#	WE#	A_9	A_0	V_{PP}	DQ_{0-15}
Read	1, 2, 3	V_{IH}	V_{IL}	V_{IL}	V_{IH}	X	X	X	D_{OUT}
Output Disable		V_{IH}	V_{IL}	V_{IH}	V_{IH}	X	X	X	High Z
Standby		V_{IH}	V_{IH}	X	X	X	X	X	High Z
Deep Power-Down	9	V_{IL}	X	X	X	X	X	X	High Z
Intelligent Identifier (Mfr)	4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IL}	X	0089H
Intelligent Identifier (Device)	4, 5	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IH}	X	4470H 4471H
Write	6, 7, 8	V_{IH}	V_{IL}	V_{IH}	V_{IL}	X	X	X	D_{IN}

Table 2. Bus Operations for BYTE-WIDE Mode (BYTE = V_{IL})

Mode	Notes	RP#	CE#	OE#	WE#	A_9	A_0	A_{-1}	V_{PP}	DQ_{0-7}	DQ_{8-14}
Read	1, 2, 3	V_{IH}	V_{IL}	V_{IL}	V_{IH}	X	X	X	X	D_{OUT}	High Z
Output Disable		V_{IH}	V_{IL}	V_{IH}	V_{IH}	X	X	X	X	High Z	High Z
Standby		V_{IH}	V_{IH}	X	X	X	X	X	X	High Z	High Z
Deep Power-Down	9	V_{IL}	X	X	X	X	X	X	X	High Z	High Z
Intelligent Identifier (Mfr)	4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IL}	X	X	89H	High Z
Intelligent Identifier (Device)	4, 5	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IH}	X	X	70H 71H	High Z
Write	6, 7, 8	V_{IH}	V_{IL}	V_{IH}	V_{IL}	X	X	X	X	D_{IN}	High Z

NOTES:
1. Refer to DC Characteristics.
2. X can be V_{IL}, V_{IH} for control pins and addresses, V_{PPL} or V_{PPH} for V_{PP}.
3. See DC Characteristics for V_{PPL}, V_{PPH}, V_{HH}, V_{ID} voltages.
4. Manufacturer and Device codes may also be accessed via a CUI write sequence. A_1–A_{17} = X.
5. Device ID = 4470H for A28F400BX-T and 4471H for A28F400BX-B.
6. Refer to Table 3 for valid D_{IN} during a write operation.
7. Command writes for Block Erase or Word/Byte Write are only executed when V_{PP} = V_{PPH}.
8. To write or erase the boot block, hold RP# at V_{HH}.
9. RP# must be at GND ±0.2V to meet the 80 µA maximum deep power-down current.

3.2 Read Operations

The 4-Mbit boot block flash family has three user read modes; Array, Intelligent Identifier, and Status Register. Status Register read mode will be discussed in detail in the "Write Operations" section.

During power-up conditions (V_{CC} supply ramping), it takes a maximum of 300 ns from when V_{CC} is at 4.5V minimum to valid data on the outputs.

3.2.1 READ ARRAY

If the memory is not in the Read Array mode, it is necessary to write the appropriate read mode command to the CUI. The 4-Mbit boot block flash family has three control functions, all of which must be logically active, to obtain data at the outputs. Chip-Enable CE# is the device selection control. Reset/Power-Down, RP# is the device power control. Output-Enable OE# is the DATA INPUT/OUTPUT (DQ[0:15] or DQ[0:7]) direction control and when active is used to drive data from the selected memory on to the I/O bus.

8

3.2.1.1 Output Control

With OE# at logic-high level (V_{IH}), the output from the device is disabled and data input/output pins (DQ[0:15] or DQ[0:7] are tri-stated. Data input is then controlled by WE#.

3.2.1.2 Input Control

With WE# at logic-high level (V_{IH}), input to the device is disabled. Data Input/Output pins (DQ[0:15] or DQ[0:7]) are controlled by OE#.

3.2.2 INTELLIGENT IDENTIFIERS

The manufacturer and device codes are read via the CUI or by taking the A_9 pin to 12V. Writing 90H to the CUI places the device into Intelligent Identifier read mode. A read of location 00000H outputs the manufacturer's identification code, 0089H, and location 00001H outputs the device code; 4470H for A28F400BX-T, 4471H for A28F400BX-B. When BYTE# is at a logic low only the lower byte of the above signatures is read and DQ_{15}/A_{-1} is a "don't care" during Intelligent Identifier mode. A read array command must be written to the memory to return to the read array mode.

3.3 Write Operations

Commands are written to the CUI using standard microprocessor write timings. The CUI serves as the interface between the microprocessor and the internal chip operation. The CUI can decipher Read Array, Read Intelligent Identifier, Read Status Register, Clear Status Register, Erase and Program commands. In the event of a read command, the CUI simply points the read path at either the array, the Intelligent Identifier, or the status register depending on the specific read command given. For a program or erase cycle, the CUI informs the write state machine that a write or erase has been requested. During a program cycle, the Write State Machine will control the program sequences and the CUI will only respond to status reads. During an erase cycle, the CUI will respond to status reads and erase suspend. After the Write State Machine has completed its task, it will allow the CUI to respond to its full command set. The CUI will stay in the current command state until the microprocessor issues another command.

The CUI will successfully initiate an erase or write operation only when V_{PP} is within its voltage range. Depending upon the application, the system designer may choose to make the V_{PP} power supply switchable, available only when memory updates are desired. The system designer can also choose to "hard-wire" V_{PP} to 12V. The 4-Mbit boot block flash family is designed to accommodate—either design practice. It is strongly recommended that RP# be tied to logical Reset for data protection during unstable CPU reset function as described in the "Product Family Overview" section.

3.3.1 BOOT BLOCK WRITE OPERATIONS

In the case of Boot Block modifications (write and erase), RP# is set to V_{HH} = 12V typically, in addition to V_{PP} at high voltage.

However, if RP# is not at V_{HH} when a program or erase operation of the boot block is attempted, the corresponding status register bit (Bit 4 for Program and Bit 5 for Erase, refer to Table 5 for Status Register Definitions) is set to indicate the failure to complete the operation.

3.3.2 COMMAND USER INTERFACE (CUI)

The Command User Interface (CUI) serves as the interface to the microprocessor. The CUI points the read/write path to the appropriate circuit block as described in the previous section. After the WSM has completed its task, it will set the WSM Status bit to a "1", which will also allow the CUI to respond to its full command set. Note that after the WSM has returned control to the CUI, the CUI will remain in its current state.

3.3.2.1 Command Set

Command Codes	Device Mode
00	Invalid/Reserved
10	Alternate Program Setup
20	Erase Setup
40	Program Setup
50	Clear Status Register
70	Read Status Register
90	Intelligent Identifier
B0	Erase Suspend
D0	Erase Resume/Erase Confirm
FF	Read Array

3.3.2.2 Command Function Descriptions

Device operations are selected by writing specific commands into the CUI. Table 3 defines the 4-Mbit boot block flash family commands.

Table 3. Command Definitions

Command	Bus Cycles Req'd	Notes	First Bus Cycle			Second Bus Cycle		
		8	Operation	Address	Data	Operation	Address	Data
Read Array	1	1	Write	X	FFH			
Intelligent Identifier	3	2, 4	Write	X	90H	Read	IA	IID
Read Status Register	2	3	Write	X	70H	Read	X	SRD
Clear Status Register	1		Write	X	50H			
Erase Setup/Erase Confirm	2	5	Write	BA	20H	Write	BA	D0H
Word/Byte Write Setup/Write	2	6, 7	Write	WA	40H	Write	WA	WD
Erase Suspend/Erase Resume	2		Write	X	B0H	Write	X	D0H
Alternate Word/Byte Write Setup/Write	2	6, 7	Write	WA	10H	Write	WA	WD

NOTES:
1. Bus operations are defined in Tables 1, and 2.
2. IA = Identifier Address: 00H for manufacturer code, 01H for device code.
3. SRD = Data read from Status Register.
4. IID = Intelligent Identifier Data.
Following the Intelligent Identifier Command, two read operations access manufacturer and device codes.
5. BA = Address within the block being erased.
6. WA = Address to be written.
WD = Data to be written at location WD.
7. Either 40H or 10H commands is valid.
8. When writing commands to the device, the upper data bus [DQ_8–DQ_{15}] = X which is either V_{CC} or V_{SS} to avoid burning additional current.

Invalid/Reserved

These are unassigned commands. It is not recommended that the customer use any command other than the valid commands specified above. Intel reserves the right to redefine these codes for future functions.

Read Array (FFH)

This single write command points the read path at the array. If the host CPU performs a CE#/OE# controlled read immediately following a two-write sequence that started the WSM, then the device will output status register contents. If the Read Array command is given after Erase Setup the device is reset to read the array. A two Read Array command sequence (FFH) is required to reset to Read Array after Program Setup.

Intelligent Identifier (90H)

After this command is executed, the CUI points the output path to the Intelligent Identifier circuits. Only Intelligent Identifier values at addresses 0 and 1 can be read (only address A_0 is used in this mode, all other address inputs are ignored).

Read Status Register (70H)

This is one of the two commands that is executable while the state machine is operating. After this command is written, a read of the device will output the contents of the status register, regardless of the address presented to the device.

The device automatically enters this mode after program or erase has completed.

Clear Status Register (50H)

The WSM can only set the Program Status and Erase Status bits in the status register, it can not clear them. Two reasons exist for operating the status register in this fashion. The first is a synchronization. The WSM does not know when the host CPU has read the status register, therefore it would not know when to clear the status bits. Secondly, if the CPU is programming a string of bytes, it may be more efficient to query the status register after programming the string. Thus, if any errors exist while programming the string, the status register will return the accumulated error status.

Program Setup (40H or 10H)

This command simply sets the CUI into a state such that the next write will load the address and data registers. Either 40H or 10H can be used for Program Setup. Both commands are included to accommodate efforts to achieve an industry standard command code set.

Program

The second write after the program setup command, will latch addresses and data. Also, the CUI initiates the WSM to begin execution of the program algorithm. While the WSM finishes the algorithm, the device will output Status Register contents. Note that the WSM cannot be suspended during programming.

Erase Setup (20H)

Prepares the CUI for the Erase Confirm command. No other action is taken. If the next command is not an Erase Confirm command then the CUI will set both the Program Status and Erase Status bits of the Status Register to a "1", place the device into the Read Status Register state, and wait for another command.

Erase Confirm (D0H)

If the previous command was an Erase Setup command, then the CUI will enable the WSM to erase, at the same time closing the address and data latches, and respond only to the Read Status Register and Erase Suspend commands. While the WSM is executing, the device will output Status Register data when OE# is toggled low. Status Register data can only be updated by toggling either OE# or CE# low.

Erase Suspend (B0H)

This command only has meaning while the WSM is executing an Erase operation, and therefore will only be responded to during an erase operation. After this command has been executed, the CUI will set an output that directs the WSM to suspend Erase operations, and then return to responding to only Read Status Register or to the Erase Resume commands. Once the WSM has reached the Suspend state, it will set an output into the CUI which allows the CUI to respond to the Read Array, Read Status Register, and Erase Resume commands. In this mode, the CUI will not respond to any other commands. The WSM will also set the WSM Status bit to a "1". The WSM will continue to run, idling in the SUSPEND state, regardless of the state of all input control pins, with the exclusion of RP#. RP# will immediately shut down the WSM and the remainder of the chip. During a suspend operation, the data and address latches will remain closed, but the address pads are able to drive the address into the read path.

Erase Resume (D0H)

This command will cause the CUI to clear the Suspend state and set the WSM Status bit to a "0", but only if an Erase Suspend command was previously issued. Erase Resume will not have any effect in all other conditions.

3.3.3 STATUS REGISTER

The 4-Mbit boot block flash family contains a status register which may be read to determine when a program or erase operation is complete, and whether that operation completed successfully. The status register may be read at any time by writing the Read Status command to the CUI. After writing this command, all subsequent Read operations output data from the status register until another command is written to the CUI. A Read Array command must be written to the CUI to return to the Read Array mode.

The status register bits are output on DQ[0:7] whether the device is in the byte-wide (x8) or word-wide (x16) mode. In the word-wide mode the upper byte, DQ[8:15] is set to 00H during a Read Status command. In the byte-wide mode, DQ[8:14] are tri-stated and DQ_{15}/A_{-1} retains the low order address function.

It should be noted that the contents of the status register are latched on the falling edge of OE# or CE# whichever occurs last in the read cycle. This prevents possible bus errors which might occur if the contents of the status register change while reading the status register. CE# or OE# must be toggled with each subsequent status read, or the completion of a program or erase operation will not be evident.

The Status Register is the interface between the microprocessor and the Write State Machine (WSM). When the WSM is active, this register will indicate the status of the WSM, and will also hold the bits indicating whether or not the WSM was successful in performing the desired operation. The WSM sets status bits "Three" through "Seven" and clears bits "Six" and "Seven", but cannot clear status bits "Three" through "Five". These bits can only be cleared by the controlling CPU through the use of the Clear Status Register command.

3.3.3.1 Status Register Bit Definition

Table 4. Status Register Definitions

WSMS	ESS	ES	PS	VPPS	R	R	R
7	6	5	4	3	2	1	0

	NOTES:
SR.7 = WRITE STATE MACHINE STATUS 1 = Ready 0 = Busy	Write State Machine Status bit must first be checked to determine byte/word program or block erase completion, before the Program or Erase Status bits are checked for success.
SR.6 = ERASE SUSPEND STATUS 1 = Erase Suspended 0 = Erase in Progress/Completed	When Erase Suspend is issued, WSM halts execution and sets both WSMS and ESS bits to "1". ESS bit remains set to "1" until an Erase Resume command is issued.
SR.5 = ERASE STATUS 1 = Error in Block Erasure 0 = Successful Block Erase	When this bit is set to "1". WSM has applied the maximum number of erase pulses to the block and is still unable to successfully perform an erase verify.
SR.4 = PROGRAM STATUS 1 = Error In Byte/Word Program 0 = Successful Byte/Word Program	When this bit is set to "1", WSM has attempted but failed to Program a byte or word.
SR.3 = V_{PP} STATUS 1 = V_{PP} Low Detect; Operation Abort 0 = V_{PP} OK	The V_{PP} Status bit unlike an A/D converter, does not provide continuous indication of V_{PP} level. The WSM interrogates the V_{PP} level only after the byte write or block erase command sequences have been entered and informs the system if V_{PP} has not been switched on. The V_{PP} Status bit is not guaranteed to report accurate feedback between V_{PPL} and V_{PPH}.
SR.2–SR.0 = RESERVED FOR FUTURE ENHANCEMENTS	These bits are reserved for future use and should be masked out when polling the Status Register.

3.3.3.2 Clearing the Status Register

Certain bits in the status register are set by the write state machine, and can only be reset by the system software. These bits can indicate various failure conditions. By allowing the system software to control the resetting of these bits, several operations may be performed (such as cumulatively programming several bytes or erasing multiple blocks in sequence). The status register may then be read to determine if an error occurred during that programming or erasure series. This adds flexibility to the way the device may be programmed or erased. To clear the status register, the Clear Status Register command is written to the CUI. Then, any other command may be issued to the CUI. Note again that before a read cycle can be initiated, a Read Array command must be written to the CUI to specify whether the read data is to come from the array, status register, or Intelligent Identifier.

3.3.4 PROGRAM MODE

Program is executed by a two-write sequence. The Program Setup command is written to the CUI followed by a second write which specifies the address and data to be programmed. The write state machine will execute a sequence of internally timed events to:

1. Program the desired bits of the addressed memory word (byte), and
2. Verify that the desired bits are sufficiently programmed.

Programming of the memory results in specific bits within a byte or word being changed to a "0".

If the user attempts to program "1"s, there will be no change of the memory cell content and no error occurs.

8

Similar to erasure, the status register indicates whether programming is complete. While the program sequence is executing, bit 7 of the status register is a "0". The status register can be polled by toggling either CE# or OE# to determine when the program sequence is complete. Only the Read Status Register command is valid while programming is active.

When programming is complete, the status bits, which indicate whether the program operation was successful, should be checked. If the programming operation was unsuccessful, Bit 4 of the status register is set to a "1" to indicate a Program Failure. If Bit 3 is set then V_{PP} was not within acceptable limits, and the WSM will not execute the programming sequence.

The status register should be cleared before attempting the next operation. Any CUI instruction can follow after programming is completed; however, it must be recognized that reads from the memory, status register, or Intelligent Identifier cannot be accomplished until the CUI is given the appropriate command. A Read Array command must first be given before memory contents can be read.

Figure 6 shows a system software flowchart for device byte programming operation. Figure 7 shows a similar flowchart for device word programming operation (A28F400BX-only).

3.3.5 ERASE MODE

Erasure of a single block is initiated by writing the Erase Setup and Erase Confirm commands to the CUI, along with the addresses A[12:17], identifying the block to be erased. These addresses are latched internally when the Erase Confirm command is issued. Block erasure results in all bits within the block being set to "1".

The WSM will execute a sequence of internally timed events to:

1. Program all bits within the block
2. Verify that all bits within the block are sufficiently programmed
3. Erase all bits within the block and
4. Verify that all bits within the block are sufficiently erased

While the erase sequence is executing, Bit 7 of the status register is a "0".

When the status register indicates that erasure is complete, the status bits, which indicate whether the erase operation was successful, should be checked. If the erasure operation was unsuccessful, Bit 5 of the status register is set to a "1" to indicate an Erase Failure. If V_{PP} was not within acceptable limits after the Erase Confirm command is issued, the WSM will not execute an erase sequence; instead, Bits of the status register is set to a "1" to indicate an Erase Failure, and Bit 3 is set to a "1" to identify that V_{PP} supply voltage was not within acceptable limits.

The status register should be cleared before attempting the next operation. Any CUI instruction can follow after erasure is completed; however, it must be recognized that reads from the memory array, status register, or Intelligent Identifier can not be accomplished until the CUI is given the appropriate command. A Read Array command must first be given before memory contents can be read.

Figure 8 shows a system software flowchart for Block Erase operation.

3.3.5.1 Suspending and Resuming Erase

Since an erase operation typically requires 1.5 to 3 seconds to complete, an Erase Suspend command is provided. This allows erase-sequence interruption in order to read data from another block of the memory. Once the erase sequence is started, writing the Erase Suspend command to the CUI requests that the Write State Machine (WSM) pause the erase sequence at a predetermined point in the erase algorithm. The status register must be read to determine when the erase operation has been suspended.

At this point, a Read Array command can be written to the CUI in order to read data from blocks other than that which is being suspended. The only other valid command at this time is the Erase Resume command or Read Status Register operation.

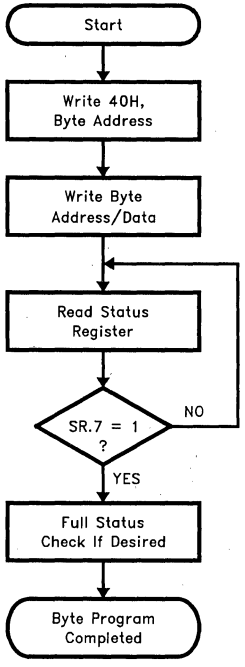
Figure 9 shows a system software flowchart detailing the operation.

During Erase Suspend mode, the chip can go into a pseudo-standby mode by taking CE# to V_{IH} and the active current is now a maximum of 10 mA. If the chip is enabled while in this mode by taking CE# to V_{IL}, the Erase Resume command can be issued to resume the erase operation.

Upon completion of reads from any block other than the block being erased, the Erase Resume command must be issued. When the Erase Resume command is given, the WSM will continue with the erase sequence and complete erasing the block. As with the end of erase, the status register must be read, cleared, and the next instruction issued in order to continue.

3.3.6 EXTENDED CYCLING

Intel has designed extended cycling capability into its ETOX III flash memory technology. The 4-Mbit boot block flash family is designed for 1,000 program/erase cycles on each of the seven blocks. The combination of low electric fields, clean oxide processing and minimized oxide area per memory cell subjected to the tunneling electric field, results in very high cycling capability.



290501-5

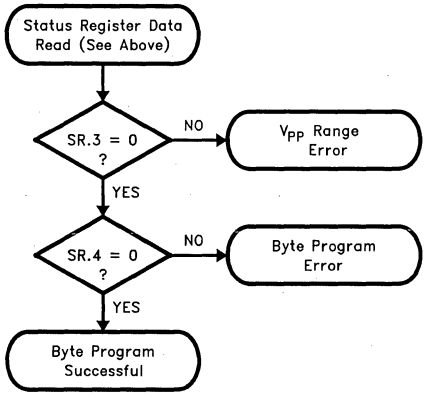
Bus Operation	Command	Comments
Write	Setup Program	Data = 40H Address = Byte to be programmed
Write	Program	Data to be programmed Address = Byte to be programmed
Read		Status Register Data. Toggle OE# or CE# to update Status Register
Standby		Check SR.7 1 = Ready, 0 = Busy

Repeat for subsequent bytes.

Full status check can be done after each byte or after a sequence of bytes.

Write FFH after the last byte programming operation to reset the device to Read Array Mode.

Full Status Check Procedure



290501-6

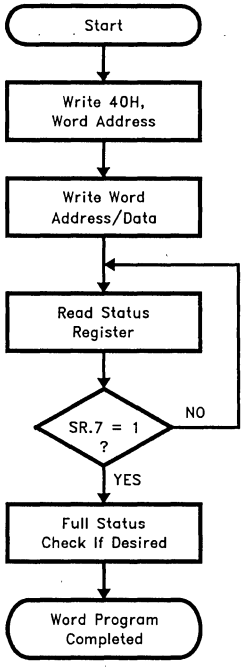
Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4 1 = Byte Program Error

SR.3 MUST be cleared, if set during a program attempt, before further attempts are allowed by the Write State Machine.

SR.4 is only cleared by the Clear Status Register Command, in cases where multiple bytes are programmed before full status is checked.

If error is detected, clear the Status Register before attempting retry or other error recovery.

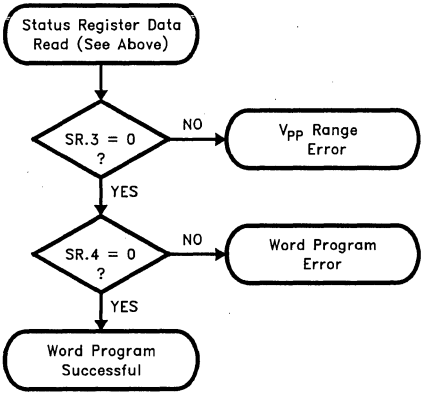
Figure 6. Automated Byte Programming Flowchart



290501-7

Bus Operation	Command	Comments
Write	Setup Program	Data = 40H Address = Word to be programmed
Write	Program	Data to be programmed Address = Word to be programmed
Read		Status Register Data. Toggle OE# or CE# to update Status Register
Standby		Check SR.7 1 = Ready, 0 = Busy
Repeat for subsequent words. Full status check can be done after each word or after a sequence of words. Write FFH after the last word programming operation to reset the device to Read Array Mode.		

Full Status Check Procedure

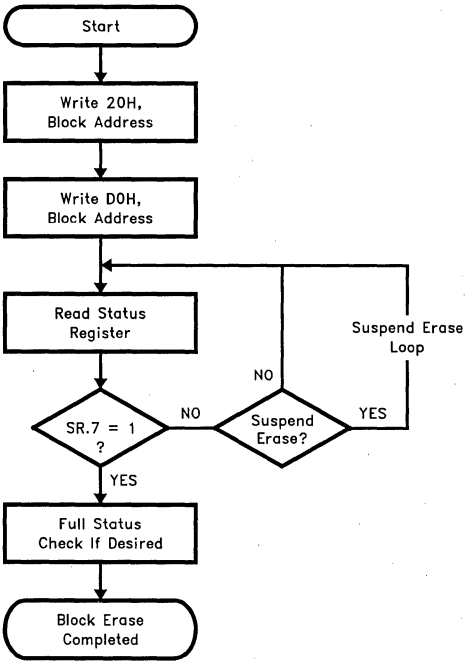


290501-8

Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4 1 = Word Program Error
SR.3 MUST be cleared, if set during a program attempt, before further attempts are allowed by the Write State Machine. SR.4 is only cleared by the Clear Status Register Command, in cases where multiple words are programmed before full status is checked. If error is detected, clear the Status Register before attempting retry or other error recovery.		

Figure 7. Automated Word Programming Flowchart

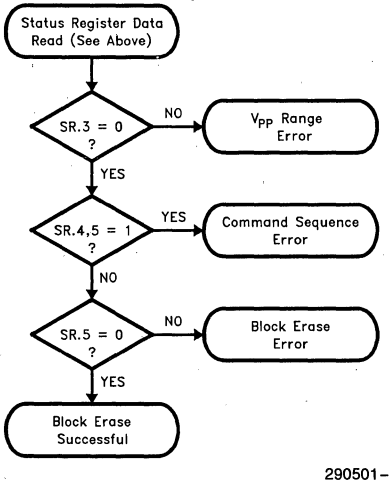
8



290501-9

Bus Operation	Command	Comments
Write	Setup Erase	Data = 20H Address = Within block to be erased
Write	Erase	Data = D0H Address = Within block to be erased
Read		Status Register Data. Toggle OE# or CE# to update Status Register
Standby		Check SR.7 1 = Ready, 0 = Busy
Repeat for subsequent blocks. Full status check can be done after each block or after a sequence of blocks. Write FFH after the last block erase operation to reset the device to Read Array Mode.		

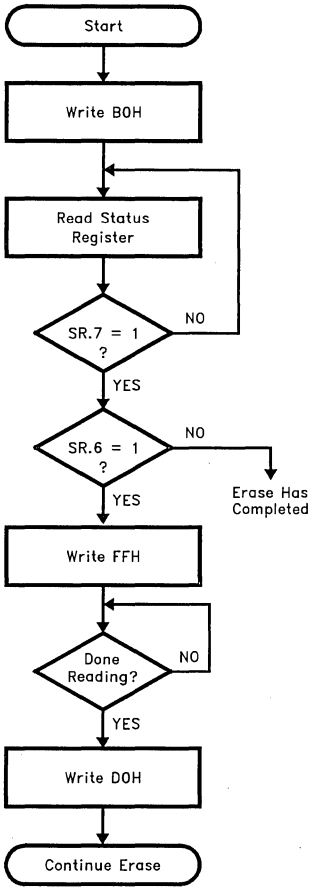
Full Status Check Procedure



290501-10

Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4,5 Both 1 = Command Sequence Error
Standby		Check SR.5 1 = Block Erase Error
SR.3 MUST be cleared, if set during an erase attempt, before further attempts are allowed by the Write State Machine. SR.5 is only cleared by the Clear Status Register Command, in cases where multiple blocks are erased before full status is checked. If error is detected, clear the Status Register before attempting retry or other error recovery.		

Figure 8. Automated Block Erase Flowchart



Bus Operation	Command	Comments
Write	Erase Suspend	Data = B0H
Read		Status Register Data. Toggle OE# or CE# to update Status Register
Standby		Check SR.7 1 = Ready
Standby		Check SR.6 1 = Suspended
Write	Read Array	Data = FFH
Read		Read array data from block other than that being erased.
Write	Erase Resume	Data = D0H

290501-11

Figure 9. Erase Suspend/Resume Flowchart

3.4 Power Consumption

3.4.1 ACTIVE POWER

With CE# at a logic-low level and RP# at a logic-high level, the device is placed in the active mode. The device I_{CC} current is a maximum 65 mA at 10 MHz with TTL input signals.

3.4.2 AUTOMATIC POWER SAVINGS

Automatic Power Savings (APS) is a low pwer feature during active mode of operation. The 4-Mbit family of products incorporate Power Reduction Control (PRC) circuitry which basically allows the device to put itself into a low current state when it is not being accessed. After data is read from the memory array, PRC logic controls the device's power consumption by entering the APS mode where maximum I_{CC} current is 3 mA and typical I_{CC} current is 1 mA. The device stays in this static state with outputs valid until a new location is read.

3.4.3 STANDBY POWER

With CE# at a logic-high level (V_{IH}), and the CUI in read mode, the memory is placed in standby mode where the maximum I_{CC} standby current is 100 µA with CMOS input signals. The standby operation disables much of the device's circuitry and substantially reduces device power consumption. The outputs (DQ[0:15] or DQ[0:7]) are placed in a high-impedance state independent of the status of the OE# signal. When the 4-Mbit boot block flash family is deselected during erase or program functions, the devices will continue to perform the erase or program function and consume program or erase active power until program or erase is completed.

8

3.4.4 DEEP POWERDOWN

The 4-Mbit boot block flash family has a RP# pin which places the device in the deep powerdown mode. When RP# is at a logic-low (GND ±0.2V), all circuits are turned off and the device typically draws a maximum 80 µA of V_{CC} current.

During read modes, the RP# pin going low deselects the memory and places the output drivers in a high impedance state. Recovery from the deep power-down state, requires a minimum of 300 ns to access valid data (t_{PHQV}).

During erase or program modes, RP# low will abort either erase or program operation. The contents of the memory are no longer valid as the data has been corrupted by the RP# function. As in the read mode above, all internal circuitry is turned off to achieve the low current level. RP# transitions to V_{IL} or turning power off to the device will clear the status register.

This use of RP# during system reset is important with automated write/erase devices. When the system comes out of reset it expects to read from the flash memory. Automated flash memories provide status information when accessed during write/erase modes. If a CPU reset occurs with no flash memory reset, proper CPU initialization would not occur because the flash memory would be providing the status information instead of array data. Intel's Flash Memories allow proper CPU initialization following a system reset through the use of RP# input. In this application RP# is controlled by the same RESET# signal that resets the system CPU.

3.5 Power-up Operation

The 4-Mbit boot block flash family is designed to offer protection against accidental block erasure or programming during power transitions. Upon power-up the 4-Mbit boot block flash family is indifferent as to which power supply, V_{PP} or V_{CC}, powers-up first. Power supply sequencing is not required.

The 4-Mbit boot block flash family ensures the CUI is reset to the read mode on power-up.

In addition, on power-up the user must either drop CE# low or present a new address to ensure valid data at the outputs.

A system designer must guard against spurious writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and CE# must be low for a command write, driving either signal to V_{IH} will inhibit writes to the device. The CUI architecture provides an added level of protection since alteration of memory contents can only occur after successful completion of the two-step command sequences. Finally the device is disabled until RP# is brought to V_{IH}, regardless of the state of its control inputs. This feature provides yet another level of memory protection.

3.6 Power Supply Decoupling

Flash memory's power switching characteristics require careful device decoupling methods. System designers are interested in 3 supply current issues:

- Standby current levels (I_{CCS})
- Active current levels (I_{CCR})
- Transient peaks produced by falling and rising edges of CE#.

Transient current magnitudes depend on the device outputs' capacitive and inductive loading. Two-line control and proper decoupling capacitor selection will suppress these transient voltage peaks. Each flash device should have a 0.1 µF ceramic capacitor connected between each V_{CC} and GND, and between its V_{PP} and GND. These high frequency, low-inherent inductance capacitors should be placed as close as possible to the package leads.

3.6.1 V_{PP} TRACE ON PRINTED CIRCUIT BOARDS

Writing to flash memories while they reside in the target system, requires special consideration of the V_{PP} power supply trace by the printed circuit board designer. The V_{PP} pin supplies the flash memory cells current for programming and erasing. One should use similar trace widths and layout considerations given to the V_{CC} power supply trace. Adequate V_{PP} supply traces and decoupling will decrease spikes and overshoots.

3.6.2 V_{CC}, V_{PP} AND RP# TRANSITIONS

The CUI latches commands as issued by system software and is not altered by V_{PP} or CE# transitions or WSM actions. Its state upon power-up, after exit from deep power-down mode or after V_{CC} transitions below V_{LKO} (Lockout voltage), is Read Array mode.

After any word/byte write or block erase operation is complete and even after V_{PP} transitions down to V_{PPL}, the CUI must be reset to Read Array mode via the Read Array command when accesses to the flash memory are desired.

ABSOLUTE MAXIMUM RATINGS*

Operating Temperature
During Read ..... −40°C to +125°C
During Block Erase
and Word/Byte Write ..... −40°C to +125°C
Temperature Under Bias ..... −40°C to +125°C

Storage Temperature ..... −65°C to +150°C

Voltage on Any Pin
(except V_{CC}, V_{PP}, A_9 and RP#)
with Respect to GND ..... −2.0V to +7.0V(2)

Voltage on Pin RP# or Pin A_9
with Respect to GND ..... −2.0V to +13.5V(2, 3)

V_{PP} Program Voltage with Respect
to GND during Block Erase
and Word/Byte Write ..... −2.0V to +14.0V(2, 3)

V_{CC} Supply Voltage
with Respect to GND ..... −2.0V to +7.0V(2)

Output Short Circuit Current..... 100 mA(4)

NOTICE: This data sheet contains information on products in the sampling and initial production phases of development. The specifications are subject to change without notice. Verify with your local Intel Sales office that you have the latest data sheet before finalizing a design.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for commercial product defined by this specification.
2. Minimum DC voltage is −0.5V on input/output pins. During transitions, this level may undershoot to −2.0V for periods <20 ns. Maximum DC voltage on input/output pins is V_{CC} + 0.5V which, during transitions, may overshoot to V_{CC} + 2.0V for periods <20 ns.
3. Maximum DC voltage on V_{PP} may overshoot to +14.0V for periods <20 ns. Maximum DC voltage on RP# or A_9 may overshoot to 13.5V for periods <20 ns.
4. Output shorted for no more than one second. No more than one output shorted at a time.

OPERATING CONDITIONS

Symbol	Parameter	Notes	Min	Max	Units
T_A	Operating Temperature		−40	125	°C
V_{CC}	V_{CC} Supply Voltage (10%)	5	4.40	5.50	V

DC CHARACTERISTICS

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Condition
I_{LI}	Input Load Current	1			±1.0	µA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or GND
I_{LO}	Output Leakage Current	1			±10	µA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or GND

8

DC CHARACTERISTICS (Continued)

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Condition
I_{CCS}	V_{CC} Standby Current	1, 3			1.5	mA	$V_{CC} = V_{CC}$ Max CE# = RP# = V_{IH}
					130	µA	$V_{CC} = V_{CC}$ Max CE# = RP# = $V_{CC} \pm 0.2V$ 28F200BX: BYTE# = $V_{CC} \pm 0.2V$ or GND
I_{CCD}	V_{CC} Deep Powerdown Current	1			80	µA	RP# = GND ±0.2V
I_{CCR}	V_{CC} Read Current for 28F400BX Byte-Wide and Word-Wide Mode	1, 5, 6			60	mA	$V_{CC} = V_{CC}$ Max, CE# = GND f = 10 MHz, I_{OUT} = 0 mA CMOS Inputs
					65	mA	$V_{CC} = V_{CC}$ Max, CE# = V_{IL} f = 10 MHz, I_{OUT} = 0 mA TTL Inputs
I_{CCW}	V_{CC} Word/Byte Write Current	1, 4			65	mA	Word Write in Progress
I_{CCE}	V_{CC} Block Erase Current	1,4			30	mA	Block Erase in Progress
I_{CCES}	V_{CC} Erase Suspend Current	1, 2		5	10	mA	Block Erase Suspended, CE# = V_{IH}
I_{PPS}	V_{PP} Standby Current	1			±10	µA	$V_{PP} \leq V_{CC}$
I_{PPD}	V_{PP} Deep PowerDown Current	1			5.0	µA	RP# = GND ±0.2V
I_{PPR}	V_{PP} Read Current	1			200	µA	$V_{PP} > V_{CC}$
I_{PPW}	V_{PP} Word Write Current	1, 4			40	mA	$V_{PP} = V_{PPH}$ Word Write in Progress
I_{PPW}	V_{PP} Byte Write Current	1, 4			30	mA	$V_{PP} = V_{PPH}$ Byte Write in Progress
I_{PPE}	V_{PP} Block Erase Current	1, 4			30	mA	$V_{PP} = V_{PPH}$ Block Erase in Progress
I_{PPES}	V_{PP} Erase Suspend Current	1			200	µA	$V_{PP} = V_{PPH}$ Block Erase Suspended
$I_{RP\#}$	RP# Current	1, 4			500	µA	RP# = V_{HH}
I_{ID}	A_9 Intelligent Identifier Current	1, 4			500	µA	$A_9 = V_{ID}$
V_{ID}	A_9 Intelligent Identifier Voltage		11.5		13.0	V	
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.0		V_{CC} + 0.5	V	
V_{OL}	Output Low Voltage				0.45	V	$V_{CC} = V_{CC}$ Min I_{OL} = 5.8 mA

DC CHARACTERISTICS (Continued)

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Condition
V_{OH}	Output High Voltage		2.4			V	$V_{CC} = V_{CC}$ Min $I_{OH} = -2.5$ mA
V_{PPL}	V_{PP} during Normal Operations	3	0.0		6.5	V	
V_{PPH}	V_{PP} during Erase/Write Operations	7	11.4	12.0	12.6	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.0			V	
V_{HH}	RP# Unlock Voltage		11.5		13.0	V	Boot Block Write/Erase

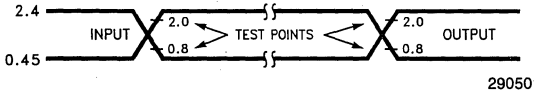
CAPACITANCE[4] $T_A = 25°C$, f = 1 MHz

Symbol	Parameter	Typ	Max	Unit	Condition
C_{IN}	Input Capacitance	6	8	pF	$V_{IN} = 0V$
C_{OUT}	Output Capacitance	10	12	pF	$V_{OUT} = 0V$

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at $V_{CC} = 5.0V$, $V_{PP} = 12.0V$, T = 25°C. These currents are valid for all product versions (packages and speeds).
2. I_{CCES} is specified with the device deselected. If the device is read while in Erase Suspend Mode, current draw is the sum of I_{CCES} and I_{CCR}.
3. Block Erases and Word/Byte Writes are inhibited when $V_{PP} = V_{PPL}$ and not guaranteed in the range between V_{PPH} and V_{PPL}.
4. Sampled, not 100% tested.
5. Automatic Power Savings (APS) reduces I_{CCR} to less than 1 mA typical in static operation.
6. CMOS Inputs are either $V_{CC} \pm 0.2V$ or GND $\pm 0.2V$. TTL Inputs are either V_{IL} or V_{IH}.
7. $V_{PP} = 12.0V \pm 5\%$ for applications requiring 1,000 block erase cycles.

STANDARD TEST CONFIGURATION

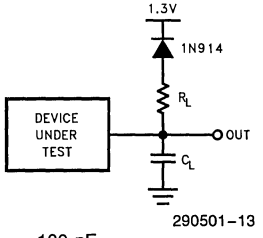
STANDARD AC INPUT/OUTPUT REFERENCE WAVEFORM



290501-12

AC test inputs are driven at V_{OH} (2.4 V_{TTL}) for a Logic "1" and V_{OL} (0.45 V_{TTL}) for a logic "0". Input timing begins at V_{IH} (2.0 V_{TTL}) and V_{IL} (0.8 V_{TTL}). Output timing ends at V_{IH} and V_{IL}. Input rise and fall times (10% to 90%) < 10 ns.

STANDARD AC TESTING LOAD CIRCUIT



290501-13

C_L = 100 pF
C_L Includes Jig Capacitance
R_L = 3.3 KΩ

8

AC CHARACTERISTICS—Read Only Operations[1]

Versions				A28F400BX-90[4, 5]		Unit
Symbol		Parameter	Notes	Min	Max	
t_{AVAV}	t_{RC}	Read Cycle Time		90		ns
t_{AVQV}	t_{ACC}	Address to Output Delay			90	ns
t_{ELQV}	t_{CE}	CE# to Output Delay			90	ns
t_{PHQV}	t_{PWH}	RP# High to Output Delay			300	ns
t_{GLQV}	t_{OE}	OE# to Output Delay	2		45	ns
t_{ELQX}	t_{LZ}	CE# to Output Low Z		0		ns
t_{EHQZ}	t_{HZ}	CE# High to Output High Z			35	ns
t_{GLQX}	t_{OLZ}	OE# to Output Low Z	3	0		ns
t_{GHQZ}	t_{DF}	OE# High to Output High Z	3		35	ns
	t_{OH}	Output Hold from Addresses, CE# or OE# Change, Whichever is First	3	0		ns
t_{ELFL} t_{ELFH}		CE# to BYTE# Switching Low or High	3		5	ns
t_{FHQV}		BYTE# Switching High to Valid Output Delay	3, 5		90	ns
t_{FLQZ}		BYTE# Switching Low to Output High Z	3		35	ns

NOTES:
1. See AC Input/Output Reference Waveform for timing measurements.
2. OE# may be delayed up to $t_{CE}-t_{OE}$ after the falling edge of CE# without impact on t_{CE}.
3. Sampled, not 100% tested.
4. See Standard Test Configuration.
5. t_{FLQV}, BYTE# switching low to valid output delay, will be equal to t_{AVQV} from the time DQ_{15}/A_{-1} becomes valid.

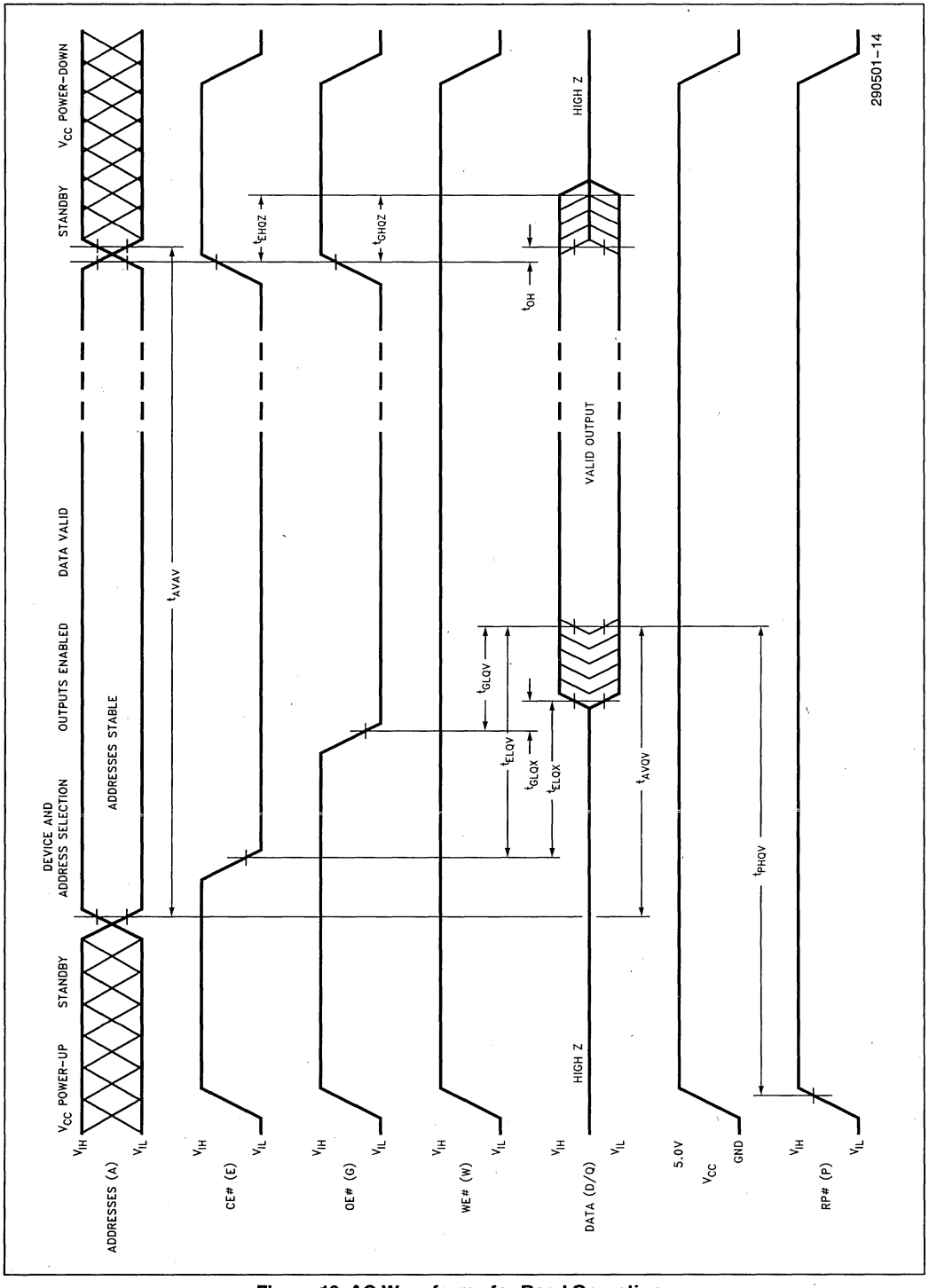


Figure 10. AC Waveforms for Read Operations

8

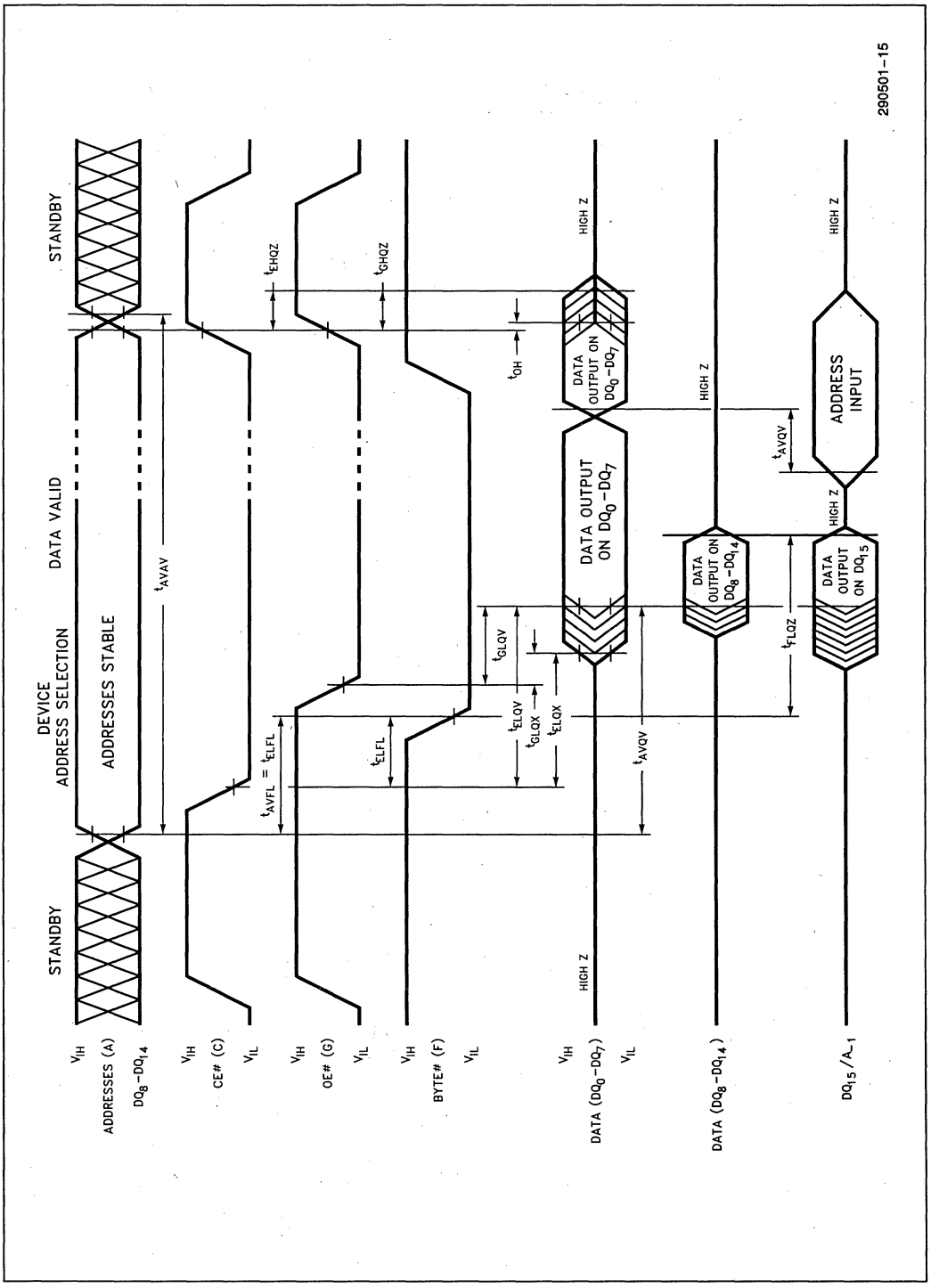


Figure 11. BYTE# Timing Diagram for Both Read and Write Operations for A28F400BX

AC CHARACTERISTICS—WE# Controlled Write Operations[1]

Versions[4]				A28F400BX-90[9]		Unit
Symbol		Parameter	Notes	Min	Max	
t_{AVAV}	t_{WC}	Write Cycle Time		90		ns
t_{PHWL}	t_{PS}	RP# High Recovery to WE# Going Low		210		ns
t_{ELWL}	t_{CS}	CE# Setup to WE# Going Low		0		ns
t_{PHHWH}	t_{PHS}	RP# V_{HH} Setup to WE# Going High	6, 8	100		ns
t_{VPWH}	t_{VPS}	V_{PP} Setup to WE# Going High	5, 8	100		ns
t_{AVWH}	t_{AS}	Address Setup to WE# Going High	3	60		ns
t_{DVWH}	t_{DS}	Data Setup to WE# Going High	4	60		ns
t_{WLWH}	t_{WP}	WE# Pulse Width		60		ns
t_{WHDX}	t_{DH}	Data Hold from WE# High	4	0		ns
t_{WHAX}	t_{AH}	Address Hold from WE# High	3	10		ns
t_{WHEH}	t_{CH}	CE# Hold from WE# High		10		ns
t_{WHWL}	t_{WPH}	WE# Pulse Width High		30		ns
t_{WHQV1}		Duration of Word/Byte Programming Operation	2, 5	7		μs
t_{WHQV2}		Duration of Erase Operation (Boot)	2, 5, 6	0.4		s
t_{WHQV3}		Duration of Erase Operation (Parameter)	2, 5	0.4		s
t_{WHQV4}		Duration of Erase Operation (Main)	2, 5	0.7		s
t_{QWL}	t_{VPH}	V_{PP} Hold from Valid SRD	5, 8	0		ns
t_{QVPH}	t_{PHH}	RP# V_{HH} Hold from Valid SRD	6, 8	0		ns
t_{PHBR}		Boot-Block Relock Delay	7, 8		100	ns

AC CHARACTERISTICS—WE# Controlled Write Operations[1] (Continued)

NOTES:
1. Read timing characteristics during write and erase operations are the same as during read-only operations. Refer to AC characteristics during Read Mode.
2. The on-chip WSM completely automates program/erase operations; program/erase algorithms are now controlled internally which includes verify and margining operations.
3. Refer to command definition table for valid A_{IN}.
4. Refer to command definition table for valid D_{IN}.
5. Program/Erase durations are measured to valid SRD data (successful operation, SR.7 = 1).
6. For Boot Block Program/Erase, RP# should be held at V_{HH} until operation completes successfully.
7. Time t_{PHBR} is required for successful relocking of the Boot Block.
8. Sampled but not 100% tested.
9. See Standard Test Configuration.

BLOCK ERASE AND WORD/BYTE WRITE PERFORMANCE V_{PP} = 12.0V ±5%

Parameter	Notes	A28F400BX-90			Unit
		Min	Typ[1]	Max	
Boot/Parameter Block Erase Time	2		1.5	10.5	s
Main Block Erase Time	2		3.0	18	s
Main Block Byte Program Time	2		1.4	5.0	s
Main Block Word Program Time	2		0.7	2.5	s

NOTES:
1. 25°C
2. Excludes System-Level Overhead.

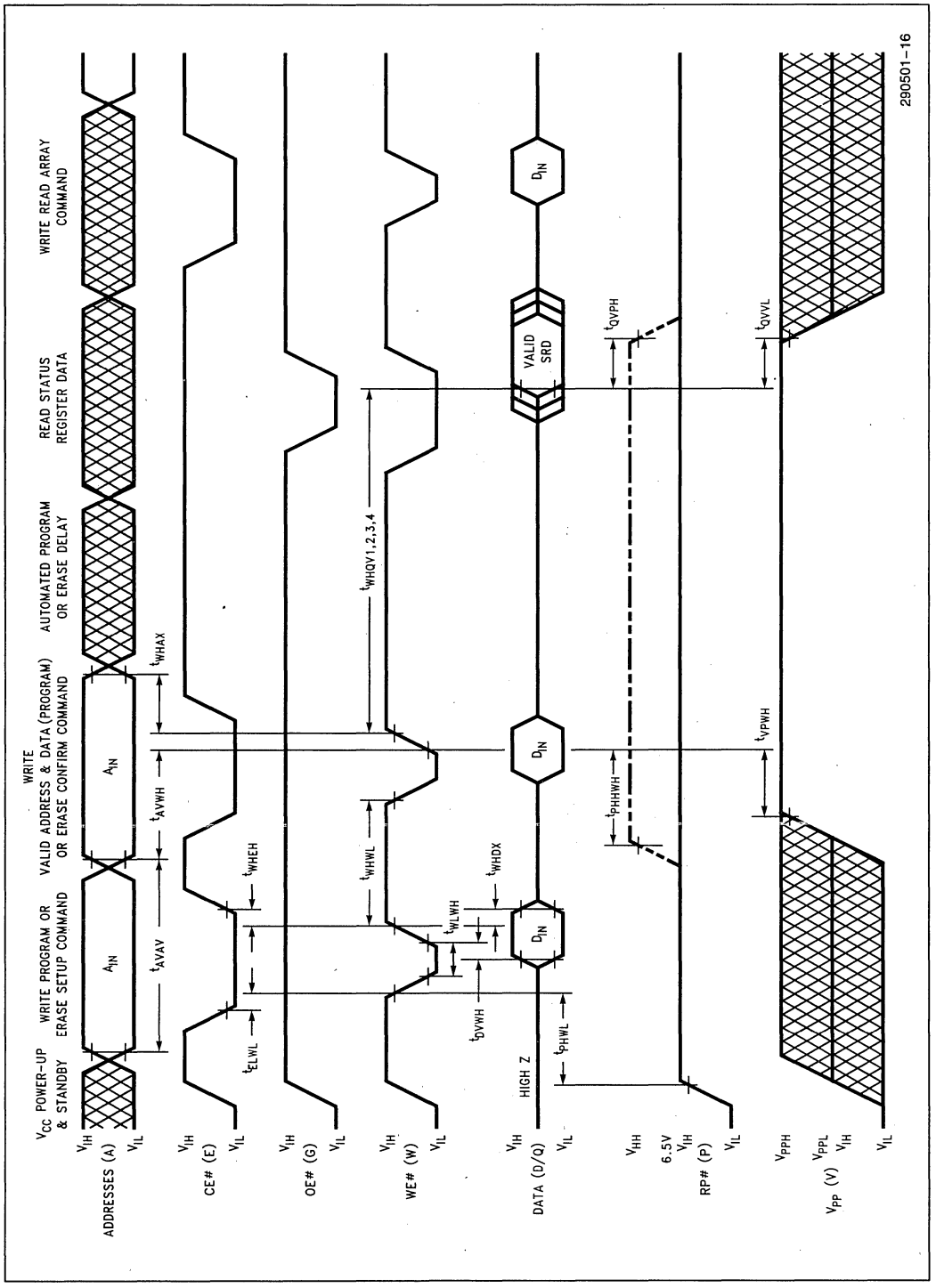


Figure 12. AC Waveforms for a Write and Erase Operations (WE#-Controlled Writes)

AC CHARACTERISTICS—CE#-CONTROLLED WRITE OPERATIONS(1, 9)

Versions				A28F400BX-90(10)		Unit
Symbol		Parameter	Notes	Min	Max	
t_{AVAV}	t_{WC}	Write Cycle Time		90		ns
t_{PHEL}	t_{PS}	RP# High Recovery to CE# Going Low		210		ns
t_{WLEL}	t_{WS}	WE# Setup to CE# Going Low		0		ns
t_{PHHEH}	t_{PHS}	RP# V_{HH} Setup to CE# Going High	6, 8	100		ns
t_{VPEH}	t_{VPS}	V_{PP} Setup to CE# Going High	5, 8	100		ns
t_{AVEH}	t_{AS}	Address Setup to CE# Going High	3	60		ns
t_{DVEH}	t_{DS}	Data Setup to CE# Going High	4	60		ns
t_{ELEH}	t_{CP}	CE# Pulse Width		60		ns
t_{EHDX}	t_{DH}	Data Hold from CE# High	4	0		ns
t_{EHAX}	t_{AH}	Address Hold from CE# High	3	10		ns
t_{EHWH}	t_{WH}	WE# Hold from CE# High		10		ns
t_{EHEL}	t_{CPH}	CE# Pulse Width High		30		ns
t_{EHQV1}		Duration of Word/Byte Programming Operation	2, 5	7		μs
t_{EHQV2}		Duration of Erase Operation (Boot)	2, 5, 6	0.4		s
t_{EHQV3}		Duration of Erase Operation (Parameter)	2, 5	0.4		s
t_{EHQV4}		Duration of Erase Operation (Main)	2, 5	0.7.		s
t_{QWL}	t_{VPH}	V_{PP} Hold from Valid SRD	5, 8	0		ns
t_{QVPH}	t_{PHH}	RP# V_{HH} Hold from Valid SRD	6, 8	0		ns
t_{PHBR}		Boot-Block Relock Delay	7		100	ns

NOTES:
1. Chip-Enable Controlled Writes: Write operations are driven by the valid combination of CE# and WE# in systems where CE# defines the write pulse-width (within a longer WE# timing waveform), all set-up, hold and inactive WE# times should be measured relative to the CE# waveform.
2, 3, 4, 5, 6, 7, 8: Refer to AC Characteristics for WE#-Controlled Write Operations.
9. Read timing characteristics during write and erase operations are the same as during read-only operations. Refer to AC Characteristics during Read Mode.
10. See Standard Test Configuration.

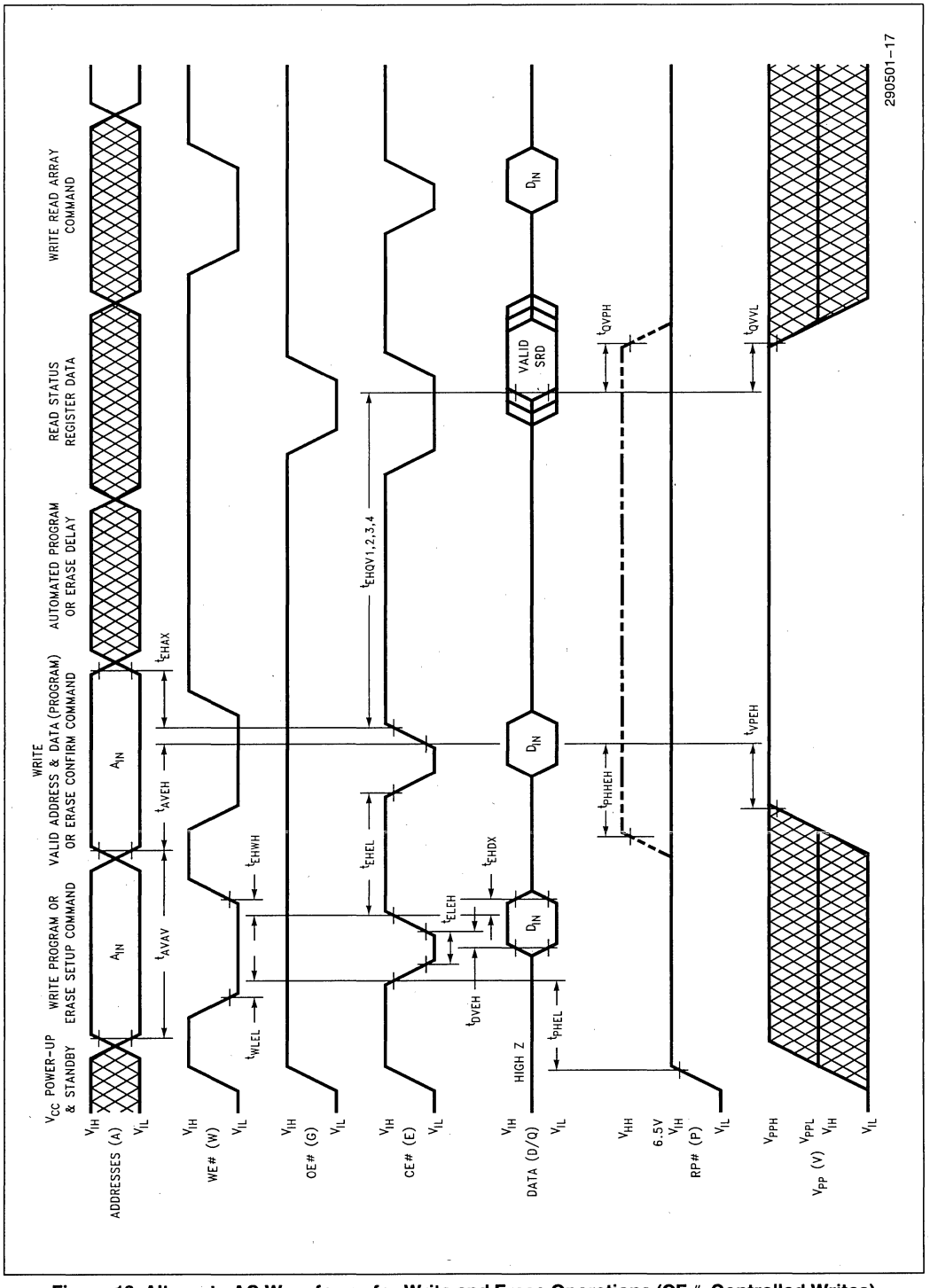
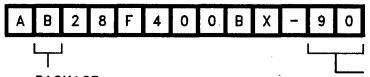


Figure 13. Alternate AC Waveforms for Write and Erase Operations (CE#-Controlled Writes)

8

ORDERING INFORMATION



290501-18

VALID COMBINATIONS:
AB28F400BX-T90
AB28F400BX-B90

ADDITIONAL INFORMATION

	Order Number
A28F200BX Datasheet	290500
28F200BX/28F002BX Datasheet	290448
28F200BX-L/28F002BX-L Datasheet	290449
28F400BX-L/28F004BX-L Datasheet	290450
AP-363 "Extended Flash BIOS Design for Portable Computers"	292098
ER-28 "ETOX™ III Flash Memory Technology"	204012
ER-29 "The Intel 2/4-MBit Boot Block Flash Memory Family"	294013

REVISION HISTORY

Number	Description
002	Changed Package Designator

A28F200BR-T/B
2-MBIT (128K x 16, 256K x 8)
SmartVoltage BOOT BLOCK FLASH MEMORY FAMILY

Automotive

- Intel SmartVoltage Technology
  - 5V or 12V Program/Erase
  - 5V Read Operation
- Very High Performance Read
  - 80 ns Max. Access Time,
  - 40 ns Max. Output Enable Time
- Low Power Consumption
  - Maximum 60 mA Read Current at 5V
- x8/x16-Selectable Input/Output Bus
  - High Performance 16- or 32-bit CPUs
- Optimized Array Blocking Architecture
  - One 16-KB Protected Boot Block
  - Two 8-KB Parameter Blocks
  - One 96-KB Main Block
  - Three 128-KB Main Blocks
  - Top or Bottom Boot Locations
- Hardware-Protection for Boot Block
- Software EEPROM Emulation with Parameter Blocks
- Automotive Temperature Operation
  - $-40°C$ to $+125°C$
- Extended Cycling Capability
  - 10,000 Block Erase Cycles for Parameter Blocks
  - 1,000 Block Erase Cycles for Main Blocks
- Automated Word/Byte Write and Block Erase
  - Industry-Standard Command User Interface
  - Status Registers
  - Erase Suspend Capability
- SRAM-Compatible Write Interface
- Automatic Power Savings Feature
  - 1 mA Typical I_{CC} Active Current in Static Operation
- Reset/Deep Power-Down Input
  - 0.2 μA I_{CC} Typical
  - Provides Reset for Boot Operations
- Hardware Data Protection Feature
  - Erase/Write Lockout during Power Transitions
- Industry-Standard Surface Mount Packaging
  - 44-Lead PSOP: JEDEC ROM Compatible
- ETOX™ IV Flash Technology

8

December 1994
Order Number: 290542-001

A28F200BR-T/B 2-MBIT (128K x 16, 256K x 8) SmartVoltage BOOT BLOCK FLASH MEMORY FAMILY

CONTENTS

PAGE

1.0 PRODUCT FAMILY OVERVIEW 8-71
- 1.1 New Features in the SmartVoltage Products 8-71
- 1.2 Main Features 8-71
- 1.3 Applications 8-72
- 1.4 Pinouts 8-72
- 1.5 Pin Descriptions 8-74

2.0 PRODUCT DESCRIPTION 8-75
- 2.1 Memory Organization 8-75
  - 2.1.1 BLOCKING 8-75

3.0 PRODUCT FAMILY PRINCIPLES OF OPERATION 8-76
- 3.1 Bus Operations 8-78
- 3.2 Read Operations 8-78
  - 3.2.1 READ ARRAY 8-78
  - 3.2.2 INTELLIGENT IDENTIFIERS 8-78
- 3.3 Write Operations 8-78
  - 3.3.1 COMMAND USER INTERFACE 8-78
  - 3.3.2 STATUS REGISTER 8-81
  - 3.3.3 PROGRAM MODE 8-81
  - 3.3.4 ERASE MODE 8-83
- 3.4 Boot Block Locking 8-86
  - 3.4.1 $V_{PP} = V_{IL}$ FOR COMPLETE PROTECTION 8-86
  - 3.4.2 WP# $= V_{IL}$ FOR BOOT BLOCK LOCKING 8-87

CONTENTS

PAGE

- 3.4.3 RP# $= V_{HH}$ OR WP# $= V_{IH}$ FOR BOOT BLOCK UNLOCKING 8-87
- 3.5 Power Consumption 8-87
  - 3.5.1 ACTIVE POWER 8-87
  - 3.5.2 AUTOMATIC POWER SAVINGS 8-87
  - 3.5.3 STANDBY POWER 8-87
  - 3.5.4 DEEP POWER-DOWN MODE 8-87
- 3.6 Power-Up Operation 8-87
  - 3.6.1 RP# CONNECTED TO SYSTEM RESET 8-88
- 3.7 Power Supply Decoupling 8-88
  - 3.7.1 V_{PP} TRACE ON PRINTED CIRCUIT BOARDS 8-88
  - 3.7.2 V_{CC}, V_{PP} AND RP# TRANSITIONS 8-88

4.0 ABSOLUTE MAXIMUM RATINGS .. 8-89

5.0 OPERATING CONDITIONS 8-90
- 5.1 V_{CC} Voltage 8-90
- 5.2 DC Characteristics 8-91
- 5.3 AC Characteristics 8-95

6.0 ADDITIONAL INFORMATION 8-101
- 6.1 Ordering Information 8-101
- 6.2 References 8-101
- 6.3 Revision History 8-101

1.0 PRODUCT FAMILY OVERVIEW

This datasheet contains the specifications for the automotive version of the 28F200BR family of boot block flash memory devices.

This device continues to offer the same functionality as earlier "BX" devices but adds the capability of performing program and erase operations with a 5V or 12V V_{PP}. Furthermore, the A28F200BR automatically senses which voltage is applied to the V_{PP} pin and adjusts its operation accordingly.

1.1 New Features in the SmartVoltage Products

The new SmartVoltage boot block flash memory family offers identical operation as the current BX/BL 12V program products, except for the differences listed below. All other functions are equivalent to current products, including signatures, write commands, and pinouts.

- WP# pin has replaced a DU pin. See Table 1 for details.
- 5V program/erase operation has been added that uses proven program and erase techniques with 5V ± 10% applied to V_{PP}.

If you are designing with existing "BX" 12V V_{PP} boot block products today, you should provide the capability in your board design to upgrade to these new SmartVoltage products.

Follow these guidelines to ensure compatibilty:

1. Connect WP# (DU on existing products) to a control signal, V_{CC} or GND.
2. If adding a switch on V_{PP} for write protection, switch to GND for complete write protection.
3. Allow for connecting 5V to V_{PP} instead of 12V, if desired.

1.2 Main Features

Intel's SmartVoltage technology provides the most flexible voltage solution in the industry. SmartVoltage provides two discrete voltage supply pins, V_{CC} for read operation, and V_{PP} for program and erase operation. Discrete supply pins allow system designers to use the optimal voltage levels for their design. For program and erase operations, 5V V_{PP} operation eliminates the need for in system voltage converters, while 12V V_{PP} operation provides faster program and erase for situations where 12V is available, such as manufacturing or designs where 12V is already available.

The 28F200 boot block flash memory family is a very high-performance, 2-Mbit (2,097,152 bit) flash memory family organized as either 256 Kwords (131,072 words) of 16 bits each or 512 Kbytes (262,144 bytes) of 8 bits each.

Separately erasable blocks, including a hardware-lockable boot block (16,384 bytes), two parameter blocks (8,192 Bytes each) and main blocks (one block of 98,304 bytes and one block of 131,072 bytes) define the boot block flash family architecture. See Figure 3 for memory maps. Each parameter block can be independently erased and programmed 10,000 times. Each main block can be erased 1,000 times.

The boot block is located at either the top (denoted by -T suffix) or the bottom (-B suffix) of the address map in order to accommodate different microprocessor protocols for boot code location. The hardware-lockable boot block provides complete code security for the kernel code required for system initialization. Locking and unlocking of the boot block is controlled by WP# and/or RP# (see Section 3.4 for details).

The Command User Interface (CUI) serves as the interface between the microprocessor or microcontroller and the internal operation of the boot block flash memory products. The internal Write State Machine (WSM) automatically executes the algorithms and timings necessary for program and erase operations, including verifications, thereby unburdening the microprocessor or microcontroller of these tasks. The Status Register (SR) indicates the status of the WSM and whether it successfully completed the desired program or erase operation.

Program and erase automation allows program and erase operations to be executed using an industry-standard two-write command sequence to the CUI. Data writes are performed in word or byte increments. Each byte or word in the flash memory can be programmed independently of other memory locations, unlike erases, which erase all locations within a block simultaneously.

The 4-Mbit SmartVoltage boot block flash memory family is also designed with an Automatic Power Savings (APS) feature which minimizes system battery current drain, allowing for very low power designs. To provide even greater power savings, the

boot block family includes a deep power-down mode which minimizes power consumption by turning most of the flash memory's circuitry off. This mode is controlled by the RP# pin and its usage is discussed in Section 3.5, along with other power consumption issues.

Additionally, the RP# pin provides protection against unwanted command writes du e to invalid system bus conditions that may occur during system reset and power-up/down sequences. Also, when the flash memory powers-up, it automatically defaults to the read array mode, but during a warm system reset, where power continues uniterrupted to the system components, the flash memory could remain in a non-read mode, such as erase. Consequently, the system Reset pin should be tied to RP# to reset the memory to normal read mode upon activation of the Reset pin.

The byte-wide or word-wide input/output is controlled by the BYTE# pin. See Table 1 for a detailed description of BYTE# operations, especially the usage of the DQ_{15}/A_{-1} pin.

The 28F200 products are available in a ROM/EPROM-compatible pinout and housed in the 44-lead PSOP (Plastic Small Outline) package.

Refer to the DC Characteristics Table, Section 5.2 for complete current and voltage specifications. Refer to the AC Characteristics Table, Section 5.3, for read, write and erase performance specificatio ns.

1.3 Applications

The 2-Mbit boot block flash memory family combines high-density, low-power, high-performance, cost-effective flash memories with blocking and hardware protection capabilities. Their flexibility and versatility reduce costs throughout the product life cycle. Flash memory is ideal for Just-In-Time production flow, reducing system inventory and costs, and eliminating component handling during the production phase.

When the product is in the end-user's hands, and updates or feature enhancements become necessary or mandatory, flash memory eliminates the need to replace an assembly. The update can be performed as part of routine maintenance operation by relatively unsophisticated technicians.

The reliability of such a field upgrade is enhanced by a hardware-protected 16-Kbyte boot block. If the protection methods are implemented in the circuit design, the boot block will be unchangeable. Locating the boot-strap code in this area assures a fail-safe recovery from a n update operation that failed to complete correctly.

The two 8-Kbyte parameter blocks allow modification of control algorithms to reflect changes in the process or device being controlled. A variety of software algorithms allow these two blocks to behave like a standard EEPROM.

Intel's boot block architecture provides a flexible voltage solution for the different design needs of various applications. The asymmetrically-blocked memory map allows the integration of several memory components into a single flash device. The boot block provides a secure boot PROM; the parameter blocks can emulate EEPROM functionality for parameter store with proper software techniques; and the main blocks provide code and data storage with access times fast enough to execute code in place, decreasing RAM requirements.

1.4 Pinouts

Intel's SmartVoltage boot block architecture provides upgrade paths in every package pinout to the 8-Mbit density. The 28F200 44-lead PSOP pinout follows the industry standard ROM/EPROM pinout as shown in Figure 2.

Pinouts for the corresponding 4-Mbit and 8-Mbit components are also provided for convenient reference. 2-Mbit pinouts are given on the chip illustration in the center, with 2-Mbit and 8-Mbit pinouts going outward from the center.

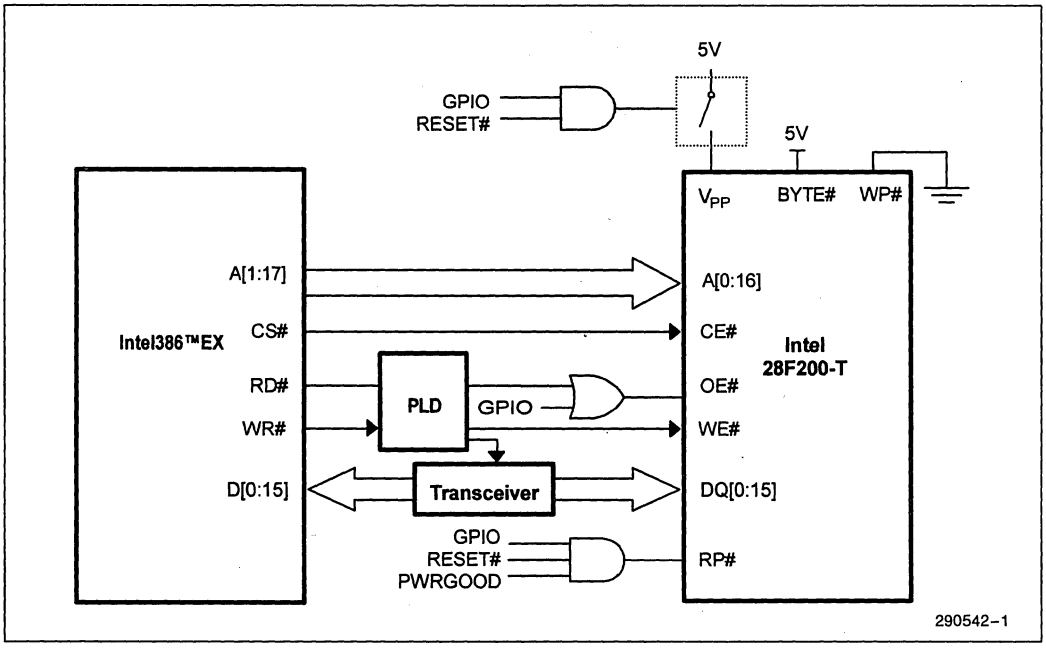
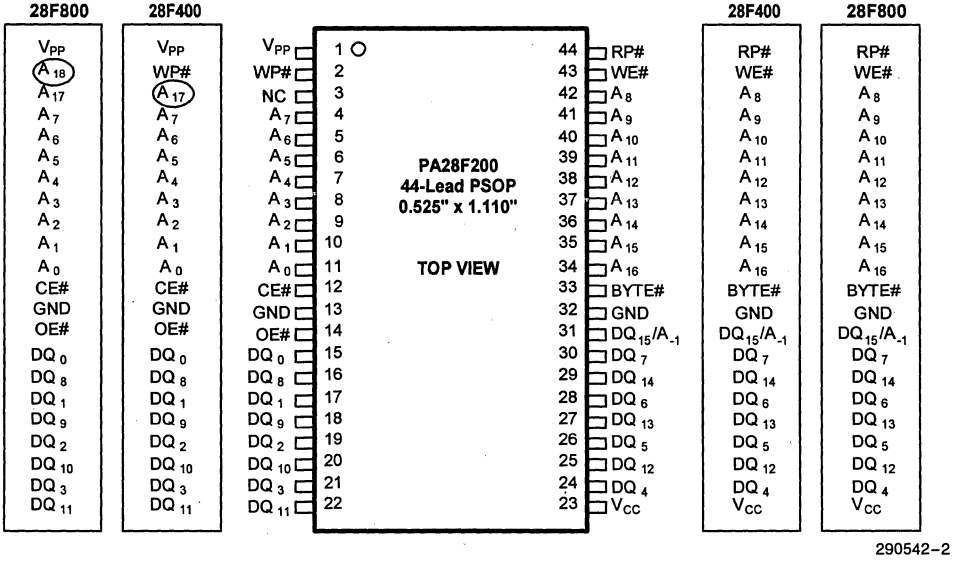


Figure 1. 28F200BX Interface to Intel386™ Microprocessor



NOTE:
Pin 2 is DU for BX 12V V_{PP} Versions.

Figure 2. 44-Lead PSOP Lead Configuration for x8/x16 28F200 Is Compatible with 4 and 8 Mbit.

8

1.5 Pin Descriptions

Table 1. 28F200 Pin Descriptions

Symbol	Type	Name and Function
A_0–A_{16}	INPUT	**ADDRESS INPUTS** for memory addresses. Addresses are internally latched during a write cycle.
A_9	INPUT	**ADDRESS INPUT:** When A_9 is at V_{HH} the signature mode is accessed. During this mode, A_0 decodes between the manufacturer and device IDs. When BYTE# is at a logic low, only the lower byte of the signatures are read. DQ_{15}/A_{-1} is a don't care in the signature mode when BYTE# is low.
DQ_0–DQ_7	INPUT/OUTPUT	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a Program command. Inputs commands to the Command User Interface when CE# and WE# are active. Data is internally latched during the Write cycle. Outputs array, Intelligent Identifier and Status Register data. The data pins float to tri-state when the chip is de-selected or the outputs are disabled.
DQ_8–DQ_{15}	INPUT/OUTPUT	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a Program command. Data is internally latched during the Write cycle. Outputs array data. The data pins float to tri-state when the chip is de-selected or the outputs are disabled as in the byte-wide mode (BYTE# = "0"). In the byte-w ide mode DQ_{15}/A_{-1} becomes the lowest order address for data output on DQ_0–DQ_7.
CE#	INPUT	**CHIP ENABLE:** Activates the device's control logic, input buffers, decoders and sense amplifiers. CE# is active low. CE# high de-selects the memory device and reduces power consumption to standby levels. If CE# and RP# are high, but not at a CMOS high level, the standby current will increase due to current flow through the CE# and RP# input stages.
OE#	INPUT	**OUTPUT ENABLE:** Enables the device's outputs through the data buffers during a read cycle. OE# is active low.
WE#	INPUT	**WRITE ENABLE:** Controls writes to the Command Register and array blocks. WE# is active low. Addresses and data are latched on the rising edge of the WE# pulse.
RP#	INPUT	**RESET/DEEP POWER-DOWN:** Uses three voltage levels (V_{IL}, V_{IH}, and V_{HH}) to control two different functions: reset/deep power-down mode and boot block unlocking. It is backwards-compatible with the 28F200BX/BL. **When RP# is at logic low, the device is in reset/deep power-down mode,** which puts the outputs at High-Z, resets the Write State Machine, and draws minimum current. **When RP# is at logic high, the device is in standard operation.** When RP# transitions from logic-low to logic-high, the device defaults to the read array mode. **When RP# is at V_{HH}, the boot block is unlocked** and can be programmed or erased. This overides any control from the WP# input.

Table 1. 28F200 Pin Descriptions (Continued)

Symbol	Type	Name and Function
WP#	INPUT	**WRITE PROTECT:** Provides a method for unlocking the boot block in a system without a 12V supply. **When WP# is at logic low, the boot block is locked,** preventing program and erase operations to the boot block. If a program or erase operation is a ttempted on the boot block when WP# is low, the corresponding status bit (bit 4 for program, bit 5 for erase) will be set in the Status Register to indicate the operation failed. **When WP# is at logic high, the boot block is unlocked** and can be programmed or erased. **NOTE:** This feature is overridden and the boot block unlocked when RP# is at V_{HH}. See Section 3.4 for details on write protection.
BYTE#	INPUT	**BYTE# ENABLE:** Controls whether the device operates in the byte-wide (x8) mode or the word (x16) mode. The BYTE# input must be controlled at CMOS levels to meet the CMOS current specification in the standby mode. When BYTE# is at logic low, the byte-wide mode is enabled. A 19-bit address is applied on A_{-1} to A_{17}, and 8 bits of data is read and written on DQ_0–DQ_7. When BYTE# is at logic high, the word-wide mode is enable. An 18-bit address is applied on A_0 to A_{17} and 16 bits of data is read and written on DQ_0–DQ_{15}.
V_{CC}		**DEVICE POWER SUPPLY:** 5.0V ± 10%
V_{PP}		**PROGRAM/ERASE POWER SUPPLY:** For erasing memory array blocks or programming data in each block, a voltage either of 5V ± 10% or 12V ± 5% must be applied to this pin. When $V_{PP} < V_{PPLK}$ all blocks are locked and protected against Program and Erase commands.
GND		**GROUND:** For all internal circuitry.
NC		**NO CONNECT:** Pin may be driven or left floating.

2.0 PRODUCT DESCRIPTION

2.1 Memory Organization

2.1.1 BLOCKING

This product family features an asymmetrically-blocked architecture enhancing system memory integration. Each block can be erased independently of the others up to 10,000 times. The block sizes have been chosen to optimize their functionality for common applications of nonvolatile storage. For the address locations of the blocks, see the memory maps in Figure 3.

2.1.1.1 Boot Block—1 x 16 KB

The boot block is intended to replace a dedicated boot PROM in a microprocessor or microcontroller-based system. The 16-Kbyte (16,384 bytes) boot block is located at either the top (denoted by -T suffix) or the bottom (-B suffix) of the address map to accommodate different microprocessor protocols for boot code location. This boot block features hardware controllable write-protection to protect the crucial microprocessor boot code from accidental erasure. The protection of the boot block is controlled using a combination of the V_{PP}, RP#, and WP# pins, as is detailed in Table 8.

2.1.1.2 Parameter Blocks—2 x 8 KB

The boot block architecture includes parameter blocks to facilitate storage of frequently updated small parameters that would normally require an EEPROM. By using software techniques, the byte-rewrite functionality of EEPROMs can be emulated. These techniques are detailed in Intel's AP-604, "Using Intel's Boot Block Flash Memory Parameter Blocks to Replace EEPROM." Each boot block component contains two parameter blocks of eight Kbytes (8,192 bytes) each. The parameter blocks are not write-protectable.

8

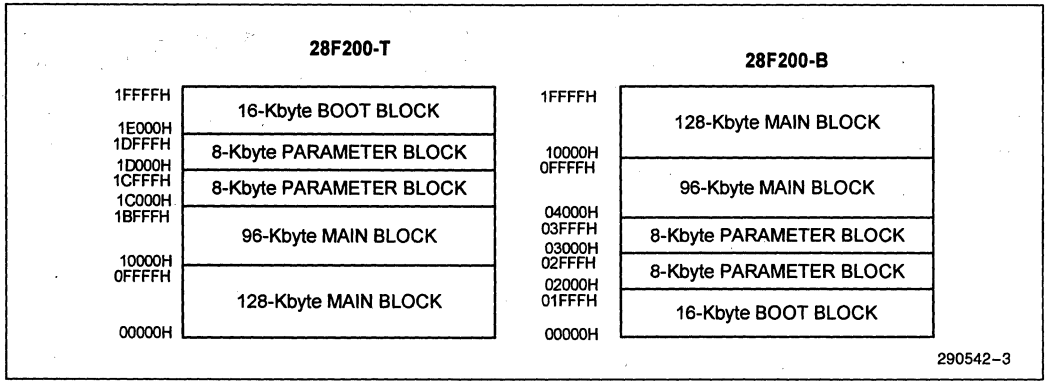


Figure 3. 28F200-T/B Memory Maps

2.1.1.3 Main Blocks—1 x 96 KB + 1 x 128 KB

After the allocation of address space to the boot and parameter blocks, the remainder is divided into main blocks for data or code storage. Each 2-Mbit device contains one 96-Kbyte (98,304 byte) block and one 128-Kbyte (131,072 byte) block. See the memory maps for each device for more information.

3.0 PRODUCT FAMILY PRINCIPLES OF OPERATION

Flash memory augments EPROM functionality with in-circuit electrical write and erase. The boot block flash family utilizes a Command User Interface (CUI) and automated algorithms to simplify write and erase operations. The CUI allows for 100% TTL-level control inputs, fixed power supplies during erasure and programming, and maximum EPROM compatibility.

When $V_{PP} < V_{PPLK}$, the device will only successfully execute the following commands: Read Array, Read Status Register, Clear Status Register and intelligent identifier mode. The device provides standard EPROM read, standby and output disable operations. Manufacturer identification and device identification data can be accessed through the CUI or through the standard EPROM A_9 high voltage access (V_{ID}) for PROM programming equipment.

The same EPROM read, standby and output disable functions are available when 5V or 12V is applied to the V_{PP} pin. In addition, 5V or 12V on V_{PP} allows write and erase of the device. All functions associated with altering memory contents: Write and Erase, Intelligent Identifier Read, and Read Status are accessed via the CUI.

The purpose of the Write State Machine (WSM) is to completely automate the write and erasure of the device. The WSM will begin operation upon receipt of a signal from the CUI and will report status back through a Status Register. The CUI will handle the WE# interface to the data and address latches, as well as system software requests for status while the WSM is in operation.

Table 2. Bus Operations for Word-Wide Mode (BYTE# = V_{IH})

Mode	Notes	RP#	CE#	OE#	WE#	A_9	A_0	V_{PP}	DQ_{0-15}
Read	1,2,3	V_{IH}	V_{IL}	V_{IL}	V_{IH}	X	X	X	D_{OUT}
Output Disable		V_{IH}	V_{IL}	V_{IH}	V_{IH}	X	X	X	High Z
Standby		V_{IH}	V_{IH}	X	X	X	X	X	High Z
Deep Power-Down	9	V_{IL}	X	X	X	X	X	X	High Z
Intelligent Identifier (Mfr)	4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IL}	X	0089 H
Intelligent Identifier (Device)	4,5	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IH}	X	See Table 4
Write	6,7,8	V_{IH}	V_{IL}	V_{IH}	V_{IL}	X	X	X	D_{IN}

Table 3. Bus Operations for Byte-Wide Mode (BYTE# = V_{IL})

Mode	Notes	RP#	CE#	OE#	WE#	A_9	A_0	A_{-1}	V_{PP}	DQ_{0-7}	DQ_{8-14}
Read	1,2,3	V_{IH}	V_{IL}	V_{IL}	V_{IH}	X	X	X	X	D_{OUT}	High Z
Output Disable		V_{IH}	V_{IL}	V_{IH}	V_{IH}	X	X	X	X	High Z	High Z
Standby		V_{IH}	V_{IH}	X	X	X	X	X	X	High Z	High Z
Deep Power-Down	9	V_{IL}	X	X	X	X	X	X	X	High Z	High Z
Intelligent Identifier (Mfr)	4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IL}	X	X	89H	High Z
Intelligent Identifier (Device)	4,5	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IH}	X	X	See Table 4	High Z
Write	6,7,8	V_{IH}	V_{IL}	V_{IH}	V_{IL}	X	X	X	X	D_{IN}	High Z

NOTES:
1. Refer to DC Characteristics.
2. X can be V_{IL}, V_{IH} for control pins and addresses, V_{PPLK} or V_{PPH} for V_{PP}.
3. See DC Characteristics for V_{PPLK}, $V_{PPH}1$, $V_{PPH}2$, V_{HH}, V_{ID} voltages.
4. Manufacturer and Device codes may also be accessed via a CUI write sequence, A_1–A_{17} = X, A_1–A_{18} = X.
5. See Table 4 of Device IDs.
6. Refer to Table 5 for valid D_{IN} during a write operation.
7. Command writes for Block Erase or Word/Byte Write are only executed when V_{PP} = $V_{PPH}1$ or $V_{PPH}2$.
8. To Write or Erase the boot block, hold RP# at V_{HH} or WP# at V_{IH}.
9. RP# must be at GND ± 0.2V to meet the maximum deep power-down current specified.

8

3.1 Bus Operations

Flash memory reads, erases and writes in-system via the local CPU. All bus cycles to or from the flash memory conform to standard microprocessor bus cycles. These bus operations are summarized in Tables 2 and 3.

3.2 Read Operations

The boot block flash device has three user read modes: array, intelligent identifier, and status register. Status register read mode will be discussed, in detail, in Section 3.3.2.

3.2.1 READ ARRAY

When RP# transitions from V_{IL} (reset) to V_{IH}, the device will be in the read array mode and will respond to the read control inputs (CE#, address inputs, and OE#) without any commands being written to the CUI.

When the device is in the read array mode, five control signals must be controlled to obtain data at the outputs.

- WE# must be logic high (V_{IH})
- CE# must be logic low (V_{IL})
- OE must be logic I ow (V_{IL})
- RP# must be logic high (V_{IH})
- BYTE# must be logic high or logic low.

In addition, the address of the desired location must be applied to the address pins. Refer to Figures 10 and 11 for the exact sequence and timing of these signals.

If the device is not in read array mode, as would be the case after a program or erase operation, the Read Mode command (FFH) must be written to the CUI before reads can take place.

3.2.1.1 Output Control

With OE# at logic-high level (V_{IH}), the output from the device is disabled and data Input/Output pins (DQ[0:15] or DQ[0:7]) are tri-stated.

3.2.1.2 Input Control

With WE# at logic-high level (V_{IH}), input to the device is disabled.

3.2.2 INTELLIGENT IDENTIFIERS

The intelligent identifiers of the SmartVoltage boot block components are identical to the boot block products that operate only at 12V V_{PP}. The manufacturer and device codes are read via the CUI or by taking the A_9 pin to V_{ID}. Writing 90H to the CUI places the device into intelligent identifier read mode. In this mode, $A_0 = 0$ outputs the manufacturer's identification code and $A_0 = 1$ outputs the device code. When BYTE# is at a logic low, only the lower byte of the above signatures is read and DQ_{15}/A_{-1} is a "don't care" during intelligent identifier mode. See the table below for product signatures. A Read Array command must be written to the memory to return to the read array mode.

Table 4. Intelligent Identifier Table

Product	Mfr. ID	Device ID	
		-T (Top Boot)	-B (Bottom Boot)
28F200	0089 H	2274 H	2275 H

3.3 WRITE OPERATIONS

3.3.1 COMMAND USER INTERFACE (CUI)

The Command User Interface (CUI) serves as the interface between the microprocessor and the internal chip controller. Commands are written to the CUI using standard microprocessor write timings. The available commands are Read Array, Read Intelligent Identifier, Read Status Register, Clear Status Register, Program and Erase (summarized in Tables 5 and 6). For Read commands, the CUI points the read path at either the array, the intelligent identifier, or the Status Register depending on the command received. For Program or Erase commands, the CUI informs the Write State Machine (WSM) that a write or erase has been requested. During the execution of a Program command, the WSM will control the programming sequences and the CUI will only respond to status reads. During an erase cycle, the CUI will respond to status reads and erase suspend. After the WSM has completed its task, it will set the WSM Status bit to a "1," which will also allow the CUI to respond to its full command set. Note that after the WSM has returned control to the CUI, the CUI will stay in the current command state until it receives another command.

Table 5. Command Set Codes and Corresponding Device Mode

Command Codes	Device Mode
00	Invalid Reserved
10	Alternate Program Set-Up
20	Erase Set-Up
40	Program Set-Up
50	Clear Status Register
70	Read Status Register
90	Intelligent Identifier
B0	Erase Suspend
D0	Erase Resume/Erase Confirm
FF	Read Array

3.3.1.1 Command Function Description

Device operations are selected by writing specific commands into the CUI. Table 5 defines the available commands.

Invalid/Reserved

These are unassigned commands and should not be used. Intel reserves the right to redefine these codes for future functions.

Read Array (FFH)

This single write cycle command points the read path at the array. If the host CPU performs a CE#/OE#-controlled Read immediately following a two-write sequence that started the WSM, then the device will output Status Register contents. If the Read Array command is given after the Erase Setup command, the device will reset to read the array. A two Read Array command sequence (FFH) is required to reset to Read Array after the Program Setup command.

Intelligent Identifier (90H)

After this command is executed, the CUI points the output path to the intelligent identifier circuits. Only intelligent identifier values at addresses 0 and 1 can be read (only address A_0 is used in this mode, all other address inputs are ignored).

Read Status Register (70H)

This is one of the two commands that is executable while the WSM is operating. After this command is written, a read of the device will output the contents of the Status Register, regardless of the address presented to the device.

The device automatically enters this mode after program or erase has completed.

Clear Status Register (50H)

The WSM can only set the Program Status and Erase Status bits in the Status Register to "1," it cannot clear them to "0."

Two reasons exist for operating the Status Register in this fashion. The first is synchronization. Since the WSM does not know when the host CPU has read the Status Register, it would not know when to clear the status bits. Secondly, if the CPU is programming a string of bytes, it may be more efficient to query the Status Register after programming the string. Thus, if any errors exist while programming the string, the Status Register will return the accumulated error status.

Table 6. Command Bus Definitions

Command	Notes	First Bus Cycle			Second Bus Cycle		
	8	Oper	Addr	Data	Oper	Addr	Data
Read Array	1	Write	X	FFH			
Intelligent Identifier	2,4	Write	X	90H	Read	IA	IID
Read Status Register	3	Write	X	70H	Read	X	SRD
Clear Status Register		Write	X	50H			
Word/Byte Write	6,7	Write	WA	40H	Write	WA	WD
Alternate Word/Byte Write	6,7	Write	WA	10H	Write	WA	WD
Block Erase/Confirm	5	Write	BA	20H	Write	BA	D0H
Erase Suspend/Resume		Write	X	B0H	Write	X	D0H

ADDRESS
BA = Block A
IA = Identifier Address
WA = Write Address
X = Don't Care

DATA
SRD = Status Register Data
IID = Identifier Data
WD = Write Data

NOTES:
1. Bus operations are defined in Tables 2 and 3.
2. IA = Identifier Address: $A_0=0$ for manufacturer code, $A_0=1$ for device code.
3. SRD - Data read from Status Register.
4. IID = Intelligent Identifier Data. Following the Intelligent Identifier command, two Read operations accessmanufacturer and device codes.
5. BA = Address within the block being erased.
6. WA = Address to be written. WD = Data to be written at location WD.
7. Either 40H or 10H commands is valid.
8. When writing commands to the device, the upper data bus [DQ_8–DQ_{15}] = X (28F200 only) which is either V_{CC} or V_{SS}, to minimize current draw.

Program Setup (40H or 10H)

This command simply sets the CUI into a state such that the next write will load the Address and Data registers. After this command is executed, the outputs default to the Status Register. A two Read Array command sequence (FFH) is required to reset to Read Array after the Program Setup command.

Program

The second write after the Program Setup command, will latch addresses and data. Also, the CUI initiates the WSM to begin execution of the program algorithm . The device outputs Status Register data when OE# is enabled. A Read Array command is required after programming, to read array data.

Erase Setup (20H)

Prepares the CUI for the Erase Confirm command. No other action is taken. If the next command is not an Erase Confirm command, then the CUI will set both the Program Status and Erase Status bits of the Status Register to a "1," place the device into the Read Status Register state, and wait for another command.

Erase Confirm (D0H)

If the previous command was an Erase Setup command, then the CUI will enable the WSM to erase, at the same time closing the address and data latches, and respond only to the Read Status Register and Erase Suspend commands. While the WSM is exe-

cuting, the device will output Status Register data when OE# is toggled low. Status Register data can only be updated by toggling either OE# or CE# low.

Erase Suspend (B0H)

This command is only valid while the WSM is executing an erase operation, and therefore will only be responded to during an erase operation. After this command has been executed, the CUI will set an output that directs the WSM to suspend erase operations, and then respond only to Read Status Register or to the Erase Resume commands. Once the WSM has reached the Suspend state, it will set an output into the CUI which allows the CUI to respond to the Read Array, Read Status Register, and Erase Resume commands. In this mode, the CUI will not respond to any other commands. The WSM will also set the WSM Status bit to a "1." The WSM will continue to run, idling in the SUSPEND state, regardless of the state of all input control pins except RP#, which will immediately shut down the WSM and the remainder of the chip, if it is made active. During a suspend operation, the data and address latches will remain closed, but the address pads are able to drive the address into the read path.

Erase Resume (D0H)

This command will cause the CUI to clear the Suspend state and clear the WSM Status Bit to a "0," but only if an Erase Suspend command was previously issued. Erase Resume will not have any effect under any other conditions.

3.3.2 STATUS REGISTER

The device contains a Status Register which may be read to determine when a program or erase operation is complete, and whether that operation completed successfully. The Status Register may be read at any time by writing the Read Status command to the CUI. After writing this command, all subsequent read operations output data from the Status Register until another command is written to the CUI. A Read Array command must be written to the CUI to return to the read array mode.

The Status Register bits are output on DQ[0:7], whether the device is in the byte-wide (x8) or word-wide (x16) mode. In the word-wide mode the upper byte, DQ[8:15], is set to 00H during a Read Status command. In the byte-wide mode, DQ[8:14] are tri-stated and DQ_{15}/A_{-1} retains the low order address function.

Important: *The contents of the Status Register are latched on the falling edge of OE# or CE#, whichever occurs last in the read cycle.* This prevents possible bus errors which might occur if the contents of the Status Register change while reading the Status Register. CE# or OE# must be toggled with each subsequent status read, or the completion of a program or erase operation will not be evident from the Status Register.

When the WSM is active, this register will indicate the status of the WSM, and will also hold the bits indicating whether or not the WSM was successful in performing the desired operation.

3.3.2.1 Clearing the Status Register

The WSM sets status bits "3" through "7" to "1," and clears bits "6" and "7" to "0," but cannot clear status bits "3" through "5" to "0." Bits 3 through 5 can only be cleared by the controlling CPU through the use of the Clear Status Register command. These bits can indicate various error conditions. By allowing the system software to control the resetting of these bits, several operations may be performed (such as cumulatively programming several bytes or erasing multiple blocks in sequence). The Status Register may then be read to determine if an error occurred during that programming or erasure series. This adds flexibility to the way the device may be programmed or erased. To clear the Status Register, the Clear Status Register command is written to the CUI. Then, any other command may be issued to the CUI. Note, again, that before a read cycle can be initiated, a Read Array command must be written to the CUI to specify whether the read data is to come from the Memory Array, Status Register, or Intelligent Identifier.

3.3.3 PROGRAM MODE

Programing is executed using a two-write sequence. The Program Setup command is written to the CUI followed by a second write which specifies the address and data to be programmed. The WSM will execute a sequence of internally timed events to:

1. Program the desired bits of the addressed memory word or byte.
2. Verify that the desired bits are sufficiently programmed.

8

Programming of the memory results in specific bits within a byte or word being changed to a "0."

If the user attempts to program "1"s, there will be no change of the memory cell content and no error occurs.

Similar to erasure, the Status Register indicates whether programming is complete. While the program sequence is executing, bit 7 of the Status Register is a "0." The Status Register can be polled by toggling either CE# or OE# to determine when the program sequence is complete. Only the Read Status Register command is valid while programming is active.

When programming is complete, the status bits, which indicate whether the program operation was successful, should be checked. If bit 3 is set to a "1," then V_{PP} was not within acceptable limits, and the WSM did not execute the programming sequence. If the program operation fails, Bit 4 of the Status Register will be set within 3.3 ms as determined by the timeout of the WSM.

The Status Register should be cleared before attempting the next operation. Any CUI instruction can follow after programming is completed; however, reads from the Memory Array, Status Register, or Intelligent Identifier cannot be accomplished until the CUI is given the Read Array command.

Table 7. Status Register Bit Definition

WSMS	ESS	ES	DWS	VPPS	R	R	R
7	6	5	4	3	2	1	0

	NOTES:
SR.7 = WRITE STATE MACHINE STATUS (WSMS) 1 = Ready 0 = Busy	Write State Machine bit must first be checked to determine Byte/Word program or Block Erase completion, before the Program or Erase Status bits are checked for success.
SR.6 = ERASE-SUSPEND STATUS (ESS) 1 = Erase Suspended 0 = Erase in Progress/Completed	When Erase Suspend is issued, WSM halts execution and sets both WSMS and ESS bits to "1." ESS bit remains set to "1" until an Erase Resume command is issued.
SR.5 = ERASE STATUS 1 = Error in Block Erasure 0 = Successful Block Erase	When this bit is set to "1," WSM has applied the maximum number of erase pulses to the block and is still unable to successfully verify block erasure.
SR.4 = PROGRAM STATUS 1 = Error in Byte/Word Program 0 = Successful Byte/Word Program	When this bit is set to "1," WSM has attempted but failed to program a byte or word.
SR.3 = V_{PP} STATUS 1 = V_{PP} Low Detect, Operation Abort 0 = V_{PP} OK	The V_{PP} Status bit, unlike an A/D converter, does not provide continuous indication of V_{PP} level. The WSM interrogates V_{PP} level only after the Bytw Write or Erase command sequences have been entered, and informs the system if V_{PP} has not been switched on. The V_{PP} Status bit is not guaranteed to report accurate feedback between V_{PPLK} and V_{PPH}.
SR.2–SR.0 = RESERVED FOR FUTURE ENHANCEMENTS	These bits are reserved for future use and should be masked out when polling the Status Register.

3.3.4 ERASE MODE

Erasure of a single block is initiated by writing the Erase Setup and Erase Confirm commands to the CUI, along with the addresses identifying the block to be erased. These addresses are latched internally when the Erase Confirm command is issued. Block erasure results in all bits within the block being set to "1."

The WSM will execute a sequence of internally timed events to:

1. Program all bits within the block to "0."
2. Verify that all bits within the block are sufficiently programmed to "0."
3. Erase all bits within the block.
4. Verify that all bits within the block are sufficiently erased.

While the erase sequence is executing, bit 7 of the Status Register is a "0."

When the Status Register indicates that erasure is complete, the status bits, which indicate whether the erase operation was successful, should be checked. If the Erase operation was unsuccessful, bit 5 of the Status Register will be set to a "1," indicating an Erase Failure. If V_{PP} was not within acceptable limits after the Erase Confirm command is issued, the WSM will not execute an erase sequence; instead, bit 5 of the Status Register is set to a "1" to indicate an Erase Failure, and bit 3 is set to a "1" to identify that V_{PP} supply voltage was not within acceptable limits.

The Status Register should be cleared before attempting the next operation. Any CUI instruction can follow after erasure is completed; however, reads from the Memory Array, Status Register, or Intelligent Identifier cannot be accomplished until the CUI is given the Read Array command.

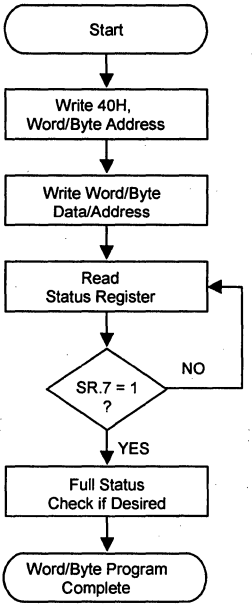
3.3.4.1 Suspending and Resuming Erase

Since an erase operation requires on the order of seconds to complete, an Erase Suspend command is provided to allow erase-sequence interruption in order to read data from another block of the memory. Once the erase sequence is started, writing the Erase Suspend command to the CUI requests that the WSM pause the erase sequence at a pre-determined point in the erase algorithm. The Status Register must then be read to determine if the erase operation has been suspended.

At this point, a Read Array command can be written to the CUI in order to read data from blocks other than that which is being suspended. The only other valid command at this time is the Erase Resume command or Read Status Register command.

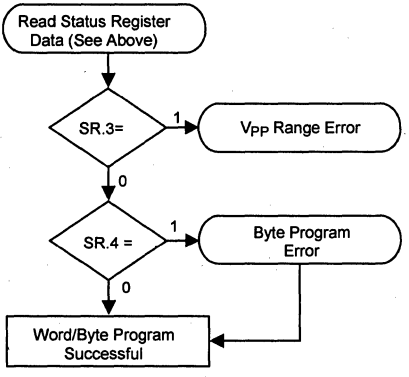
During erase suspend mode, the chip can go into a pseudo-standby mode by taking CE# to V_{IH}, which reduces active current draw.

To resume the erase operation, the chip must be enabled by taking CE# to V_{IL}, then issuing the Erase Resume command. When the Erase Resume command is given, the WSM will continue with the erase sequence and complete erasing the block. As with the end of a standard erase operation, the Status Register must be read, cleared, and the next instruction issued in order to continue.



Bus Operation	Command	Comments
Write	Setup Program	Data = 40H Addr = Word/Byte to Program
Write	Program	Data = Data to Program Addr = Location to Program
Read		Status Register Data Toggle CE# or OE# to Update SRD.
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy
Repeat for subsequent Word/Byte Writes. SR Full Status Check can be done after each Word/Byte Write, or after a sequence of Word/Byte Writes. Write FFH after the last write operation to reset device to read array mode.		

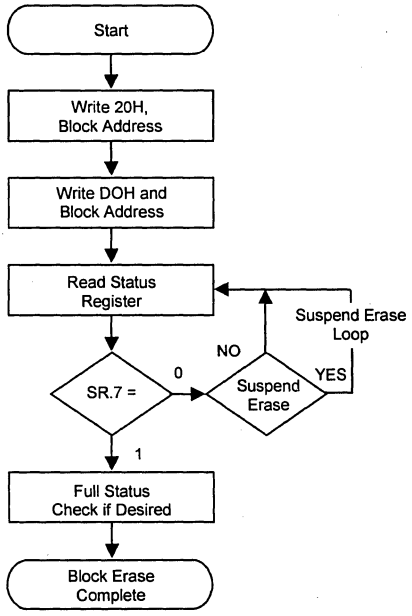
FULL STATUS CHECK PROCEDURE



Bus Operation	Command	Comments
Standby		Check SR.3 1 = VPP Low Detect
Standby		Check SR.4 1 = VPP Byte Program Error
SR.3 MUST be cleared, if set during a program attempt, before further attempts are allowed by the Write State Machine. SR.4 is only clear by the Clear Status Register Command, in cases where multiple bytes are programmed before full status is checked. If error is detected, clear the Status Register before attempting retry or other error recovery.		

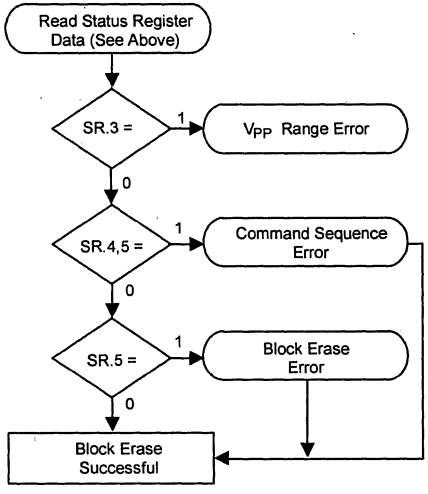
290542-4

Figure 4. Automated Word/Byte Programming Flowchart



Bus Operation	Command	Comments
Write	Erase Setup	Data = 20H Addr = Within Block to be Erased
Write	Erase Confirm	Data = D0H Addr = Within Block to be Erased
Read		Status Register Data Toggle CE# or OE# to Update Status Register
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy
Repeat for subsequent block erasures. Full Status Check can be done after each block erase, or after a sequence of block erasures. Write FFH after the last operation to reset device to read array mode.		

FULL STATUS CHECK PROCEDURE

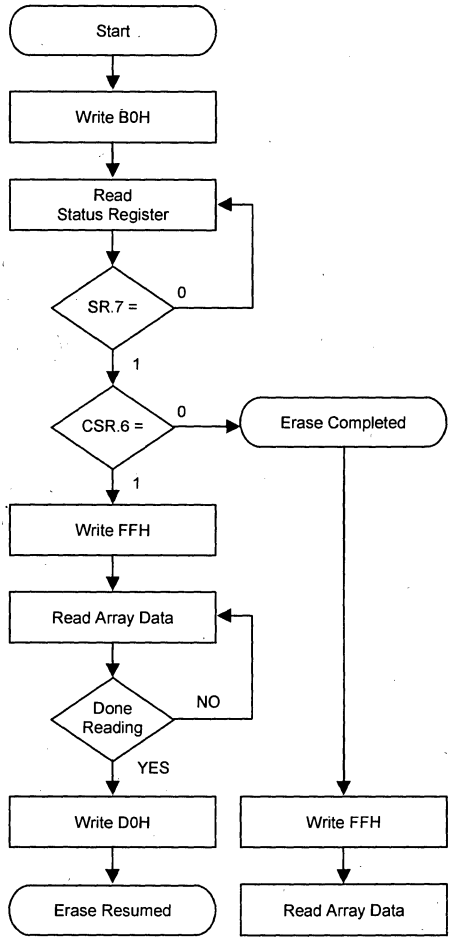


Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4,5 Both 1 = Command Sequence Error
Standby		Check SR.5 1 = Block Erase Error
SR.3 MUST be cleared, if set during an erase attempt, before further attempts are allowed by the Write State Machine. SR.5 is only clear by the Clear Status Register Command, in cases where multiple blocks are erase before full status is checked. If error is detected, clear the Status Register before attempting retry or other error recovery.		

290542-5

Figure 5. Automated Block Erase Flowchart

8



Bus Operation	Command	Comments
Write	Erase Suspend	Data = B0H Addr = X
Read		Status Register Data Toggle CE# or OE# to update SRD. Addr = X
Standby		Check SR.7 1 = WSM Ready 0 = WSM Busy
Standby		Check SR.6 1 = Erase Suspended 0 = Erase Completed
Write	Read Array	Data = FFH Addr = X
Read		Read array data from block other than the one being erased.
Write	Erase Resume	Data = D0H Addr = X

290542-6

Figure 6. Erase Suspend/Resume Flowchart

3.4 Boot Block Locking

The boot block family architecture features a hardware-lockable boot block so that the kernel code for the system can be kept secure while the parameter and main blocks are programmed and erased independently as necessary. Only the boot block can be locked independently from the other blocks.

3.4.1 $V_{PP} = V_{IL}$ FOR COMPLETE PROTECTION

For complete write protection of all blocks in the flash device, the V_{PP} programming voltage can be held low. When V_{PP} is below V_{PPLK}, any program or erase operation will result in a error in the Status Register.

3.4.2 WP# = V_{IL} FOR BOOT BLOCK LOCKING

When WP# = V_{IL}, the boot block is locked and any program or erase operation will result in an error in the Status Register. All other blocks remain unlocked in this condition and can be programmed or erased normally. Note that this feature is overridden and the boot block unlocked when RP# = V_{HH}.

3.4.3 RP# = V_{HH} OR WP# = V_{IH} FOR BOOT BLOCK UNLOCKING

Two methods can be used to unlock the boot block:

1. WP# = V_{IH}
2. RP# = V_{HH}

If both or either of these two conditions are met, the boot block will be unlocked and can be programmed or erased. The Truth Table, Table 8, clearly defines the write protection methods.

Table 8. Write Protection Truth Table for SmartVoltage Boot Block Family

V_{PP}	RP#	WP#	Write Protection Provided
V_{IL}	X	X	All Blocks Locked
$\geq V_{PPLK}$	V_{IL}	X	All Blocks Locked (Reset)
$\geq V_{PPLK}$	V_{HH}	X	All Blocks Unlocked
$\geq V_{PPLK}$	V_{IH}	V_{IL}	Boot Block Locked
$\geq V_{PPLK}$	V_{IH}	V_{IH}	All Blocks Unlocked

3.5 Power Consumption

3.5.1 ACTIVE POWER

With CE# at a logic-low level and RP# at a logic-high level, the device is placed in the active mode. Refer to the DC Characteristics table for I_{CC} current values.

3.5.2 AUTOMATIC POWER SAVINGS (APS)

Automatic Power Savings (APS) is a low-power feature during active mode of operation. The boot block flash memory family incorporates Power Reduction Control (PRC) circuitry which allows the device to put itself into a low current state when it is not being accessed. After data is read from the memory array, PRC logic controls the device's power consumption by entering the APS mode where typical I_{CC} current is less than 1 mA. The device stays in this static state with outputs valid until a new location is read.

3.5.3 STANDBY POWER

With CE# at logic-high level (V_{IH}), and the CUI in read mode, the memory is placed in standby mode. The standby operation disables much of the device's circuitry and substantially reduces device power consumption. The outputs (DQ[0:15] or DQ[0:7]) are placed in a high-impedance state independent of the status of the OE# signal. When CE# is at logic-high level during erase or program functions, the devices will continue to perform the erase or program function and consume erase or program active power until erase or program is completed.

3.5.4 DEEP POWER-DOWN MODE

The SmartVoltage boot block family supports a low typical I_{CC} in deep power-down mode. The device has a RP# pin which places the device in the deep power-down mode. When RP# is at a logic-low (GND ± 0.2V), all circuits are turned off in order to save power. (Note: BYTE# pin must be at CMOS levels to achieve the most deep power-down current savings.)

During read modes, the RP# pin going low de-selects the memory and places the output drivers in a high impedance state. Recovery from the deep power-down state, requires a minimum access time of t_{PHQV}. (See the AC Characteristics table for specification numbers.)

During erase or program modes, RP# low will abort either Erase or Program operation. The contents of the memory are no longer valid as the data has been corrupted by the RP# function. As in the read mode above, all internal circuitry is turned off to achieve the power savings.

RP# transitions to V_{IL}, or turning power off to the device will clear the Status Register.

3.6 Power-Up Operation

The device is designed to offer protection against accidental block erasure or programming during power transitions. Upon power-up, the device is indifferent as to which power supply, V_{PP} or V_{CC}, powers-up first. Power supply sequencing is not required.

A system designer must guard against spurious writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and CE# must be low for a command write, driving either signal to V_{IH} will inhibit writes to the device. The CUI architecture provides an added level of protection since alteration of memory contents can only occur after successful completion of the two-step command sequences. Finally the device is disabled until RP# is brought to V_{IH}, regardless of the state of its control inputs. By holding the device in reset (RP# connected to system PowerGood) during power up/down, invalid bus conditions that may occur can be masked. This feature provides yet another level of memory protection.

3.6.1 RP# CONNECTED TO SYSTEM RESET

The use of RP# during system reset is important with automated write/erase devices. When the system comes out of reset it expects to read from the flash memory. Automated flash memories provide status information when accessed during writes/erase modes. If a CPU reset occurs with no flash memory reset, proper CPU initialization would not occur because the flash memory would be providing the status information instead of array data. Intel's Flash memories allow proper CPU initialization following a system reset through the use of the RP# input. In this application RP# is controlled by the same RESET# signal that resets the system CPU.

3.7 Power Supply Decoupling

Flash memory's power switching characteristics require careful device decoupling methods. System designers should consider three supply current issues:

1. Standby current levels (I_{CCS})
2. Active current levels (I_{CCR})
3. Transient peaks produced by falling and rising edges of CE#.

Transient current magnitudes depend on the device outputs' capacitive and inductive loading. Two-line control and proper decoupling capacitor selection will suppress these transient voltage peaks. Each flash device should have a 0.1 μF ceramic capacitor connected between each V_{CC} and GND, and between its V_{PP} and GND. These high frequency, inherently low inductance capacitors should be placed as close as possible to the package leads.

3.7.1 V_{PP} TRACE ON PRINTED CIRCUIT BOARDS

Writing to flash memories while they reside in the target system, requires special consideration of the V_{PP} power supply trace by the printed circuit board designer. The V_{PP} pin supplies the flash memory cells current for programming and erasing. One should use similar trace widths and layout considerations given to the V_{CC} power supply trace. Adequate V_{PP} supply traces, and decoupling capacitors placed adjacent to the component, will decrease spikes and overshoots.

3.7.2 V_{CC}, V_{PP} AND RP# TRANSITIONS

The CUI latches commands as issued by system software and is not altered by V_{PP} or CE# transitions or WSM actions. Its default state upon power-up, after exit from deep power-down mode, or after V_{CC} transitions above V_{LKO} (Lockout voltage), is read array mode.

After any Word/Byte Write or Block Erase operation is complete and even after V_{PP} transitions down to V_{PPLK}, the CUI must be reset to read array mode via the Read Array command when accesses to the flash memory are desired.

4.0 ABSOLUTE MAXIMUM RATINGS*

Operating Temperature

During Read ..... −40°C to +125°C

During Block Erase and Word/Byte Write ..... −40°C to +125°C

Temperature Bias ..... −40°C to +125°C

Storage Temperature ..... −65°C to +125°C

Voltage on Any Pin (except V_{CC}, V_{PP}, A_9 and RP#) with Respect to GND ..... −2.0V to +7.0V(1)

Voltage on Pin RP# or Pin A_9 with Respect to GND..... −2.0V to +13.5V(1,2)

V_{PP} Program Voltage with Respect to GND during Block Erase and Word/Byte Write ..... −2.0V to +14.0V(1,2)

V_{CC} Supply Voltage with Respect to GND ..... −2.0V to +7.0V(1)

Output Short Circuit Current..... 100 mA(3)

NOTICE: This data sheet contains information on products in the sampling and initial production phases of development. The specifications are subject to change without notice. Verify with your local Intel Sales office that you have the latest data sheet before finalizing a design.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:

1. Minimum DC voltage is −0.5V on input/output pins. During transitions, this level may undershoot to −2.0V for periods <20 ns. Maximum DC voltage on input/output pins is V_{CC} + 0.5V which, during transitions, may overshoot to V_{CC} + 2.0V for periods <20 ns.
2. Maximum DC voltage on V_{PP} may overshoot to +14.0V for periods <20 ns. Maximum DC voltage on RP# or A_9 may overshoot to 13.5V for periods <20 ns.
3. Output shorted for no more than one second. No more than one output shorted at a time.

8

5.0 OPERATING CONDITIONS

Table 9. Temperature and V_{CC} Operating Conditions

Symbol	Parameter	Notes	Min	Max	Units
T_A	Operating Temperature		−40	+125	°C
V_{CC}	V_{CC} Supply Voltage (10%)		4.50	5.50	Volts

5.1 V_{CC} Voltages

Table 10. V_{CC} Supply Timing

Symbol	Parameter	Notes	Min	Max	Unit
T_{5VPH}	V_{CC} at 4.5V (minimum) to RP# High	1	2		μs

NOTE:
1. The T_{5VPH} time must be strictly followed to guarantee all other read and write specifications.

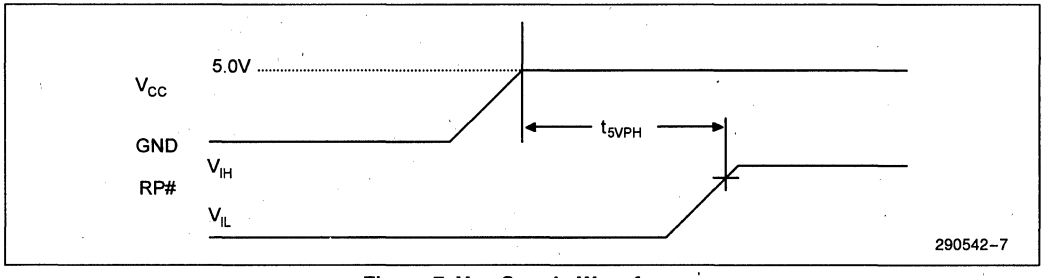


Figure 7. V_{CC} Supply Waveform

5.2 DC Characteristics

Table 11. DC Characteristics: Automotive Temperature Operation

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Conditions
I_{IL}	Input Load Current	1			± 1.0	µA	$V_{CC} = V_{CC}$ MAX $V_{IN} = V_{CC}$ or GND
I_{LO}	Output Leakage Current	1			± 10	µA	$V_{CC} = V_{CC}$ MAX $V_{IN} = V_{CC}$ or GND
I_{CCS}	V_{CC} Standby Current	1,3		0.8	2.5	mA	$V_{CC} = V_{CC}$ MAX CE# = RP# = BYTE# = V_{IH}
				70	150	µA	$V_{CC} = V_{CC}$ MAX CE# = RP# = WP# = V_{CC} ± 0.2V
I_{CCD}	V_{CC} Deep Power-Down Current	1		0.2	10	µA	$V_{CC} = V_{CC}$ MAX $V_{IN} = V_{CC}$ or GND RP# = GND ± 0.2V
I_{CCR}	V_{CC} Read Current for Word or Byte	1,5,6		50	65	mA	**CMOS** $V_{CC} = V_{CC}$ MAX CE# = V_{IL} f = 10 MHz (5V) 5 MHz (3.3V) I_{OUT} = 0 mA Inputs = GND ± 0.2V or V_{CC} ± 0.2V
				55	70	mA	**TTL** $V_{CC} = V_{CC}$ MAX CE# = V_{IL} f = 10 MHz I_{OUT} = 0 mA Inputs = V_{IL} or V_{IH}
I_{CCW}	V_{CC} Write Current for Word or Byte	1,4		25	50	mA	Word/Byte Program in Progress $V_{PP} = V_{PPH}1$ (at 5V)
				20	45	mA	Word/Byte Program in Progress $V_{PP} = V_{PPH}2$ (at 12V)

8

Table 11. DC Characteristics: Automotive Temperature Operation (Continued)

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Conditions
I_{CCE}	V_{CC} Erase Current	1,4		22	45	mA	Block Erase in Progress $V_{PP} = V_{PPH}1$ (at 5V)
				18	40	mA	Block Erase in Progress $V_{PP} = V_{PPH}2$ (at 12V)
I_{CCES}	V_{CC} Erase Suspend Current	1,2		5	12.0	mA	CE# = V_{IH} Block Erase Suspend $V_{PP} = V_{PPH}1$ (at 5V)
I_{PPS}	V_{PP} Standby Current	1		± 5	± 15	µA	$V_{PP} \leq V_{CC}$
I_{PPD}	V_{PP} Deep Power-down Current	1		0.2	10	µA	RP# = GND ± 0.2V
I_{PPR}	V_{PP} Read Current	1		50	200	µA	$V_{PP} > V_{CC}$
I_{PPW}	V_{PP} Write Current for Word/Byte	1		13	30	mA	$V_{PP} = V_{PPH}$ Word Write in Progress $V_{PP} = V_{PPH}1$ (at 5V)
				8	25	mA	$V_{PP} = V_{PPH}$ Word Write in Progress $V_{PP} = V_{PPH}2$ (at 12V)
I_{PPE}	V_{PP} Erase Current	1		15	25	mA	$V_{PP} = V_{PPH}$ Block Erase in Progress $V_{PP} = V_{PPH}1$ (at 5V)
				10	20	mA	$V_{PP} = V_{PPH}$ Block Erase in Progress $V_{PP} = V_{PPH}2$ (at 12V)
I_{PPES}	V_{PP} Erase Suspend Current	1		50	200	µA	$V_{PP} = V_{PPH}$ Block Erase Suspend in Progress
$I_{RP\#}$	RP# Boot Block Unlock Current	1,4			500	µA	RP# = V_{HH} V_{PP} = 12V
I_{ID}	A_9 Intelligent Identifier Current	1,4			500	µA	$A_9 = V_{ID}$
V_{ID}	A_9 Intelligent Identifier Voltage		11.4		12.6	V	
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.0		V_{CC} ±0.5V	V	

Table 11. DC Characteristics: Automotive Temperature Operation (Continued)

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Conditions
V_{OL}	Output Low Voltage (TTL)				0.45	V	$V_{CC} = V_{CC}$ MIN $I_{OL} = 5.8$ mA $V_{PP} = 12V$
$V_{OH}1$	Output High Voltage (TTL)		2.4			V	$V_{CC} = V_{CC}$ MIN $I_{OH} = -1.5$ mA
$V_{OH}2$	Output High Voltage (CMOS)		V_{CC} − 0.4V			V	$V_{CC} = V_{CC}$ MIN $I_{OH} = -100\ \mu A$
V_{PPLK}	V_{PP} Lock-Out Voltage	3	0.0		1.5	V	Complete Write Protection
$V_{PPH}1$	V_{PP} (Program/ Erase Operations)	7,8	4.5		5.5	V	V_{PP} at 5V
$V_{PPH}2$	V_{PP} (Program/ Erase Operations)	7,8	11.4		12.6	V	V_{PP} at 12V
V_{LKO}	V_{CC} Erase/Write Lock Voltage	10	3.9			V	$V_{PP} = 12V$
V_{HH}	RP# Unlock Voltage		11.4		12.6	V	Boot Block Write/Erase $V_{PP} = 12V$

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at $V_{CC} = 5.0V$, $T = 25°C$. These currents are valid for all product versions (packages and speeds).
2. I_{CCES} is specified with the device de-selected. If the device is read while in erase suspend mode, current draw is the sum of I_{CCES} and I_{CCR}.
3. Block erases and word/byte writes are inhibited when $V_{PP} = V_{PPLK}$, and not guaranteed in the range between $V_{PPH}1$ and V_{PPLK}.
4. Sampled, not 100% tested.
5. Automatic Power Savings (APS) reduces I_{CCR} to less than 1 mA typical, in static operation.
6. CMOS Inputs are either $V_{CC} \pm 0.2V$ or GND ± 0.2V. TTL Inputs are either V_{IL} or V_{IH}.

8

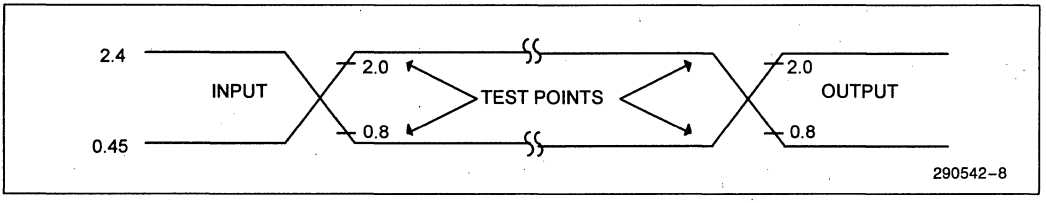
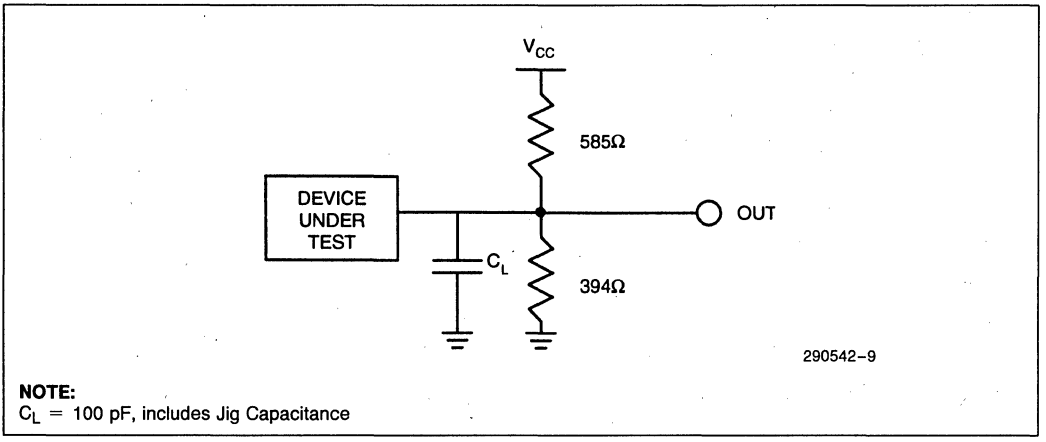


Figure 8. 5V Inputs and Measurement Points



NOTE:
C_L = 100 pF, includes Jig Capacitance

Figure 9. 5V Standard Test Configuration

5.3 AC Characteristics

Table 12. AC Characteristics: Read Only Operations[1] (Automotive Temperature)

Symbol	Parameter	Note	Min	Max	Unit
t_{AVAV}	Read Cycle Time		80		ns
t_{AVQV}	Address to Output Delay			80	ns
t_{ELQV}	CE# to Output Delay	2		80	ns
t_{PHQV}	RP# to Output Delay			500	ns
t_{GLQV}	OE# to Output Delay	2		40	ns
t_{ELQX}	CE# to Output in Low Z	3	0		ns
t_{EHQZ}	CE# to Output in High Z	3		30	ns
t_{GLQX}	OE# to Output in Low Z	3	0		ns
t_{GHQZ}	OE# to Output in High Z	3		30	ns
t_{OH}	Output Hold from Address CE#, or OE# Change Whichever Occurs First	3	0	0	ns
t_{ELFL} t_{ELFH}	CE# Low to BYTE# High or Low	3		5	ns
t_{AVFL}	Address to BYTE# High or Low	3		5	ns
t_{FLQV} t_{FHQV}	BYTE# to Output Delay	3,4		80	ns
t_{FLQZ}	BYTE# Low to Output in High Z	3		30	ns

NOTES:
1. See AC Input/Output Reference Waveform for timing measurements.
2. OE# may be delayed up to $t_{CE}-t_{OE}$ after the falling edge of CE# without impact on t_{CE}.
3. Sampled, but not 100% tested.
4. t_{FLQV}, BYTE# switching low to valid output delay will be equal to t_{AVQV}, measured from the time DQ_{15}/A_{-1} becomes valid.
5. See 5V High Speed Test Configuration. (Figure 9)

8

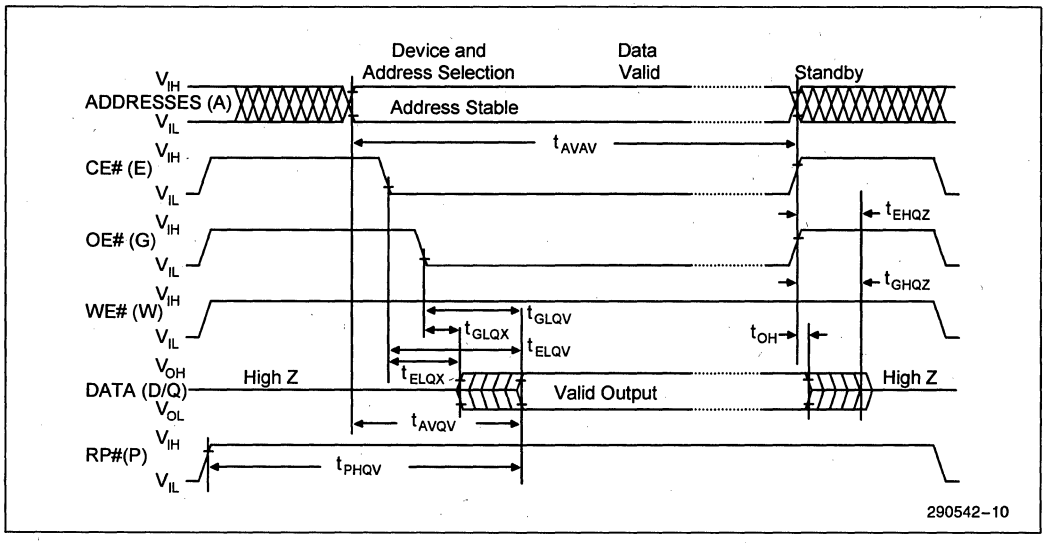


Figure 10. AC Waveforms for Read Operations

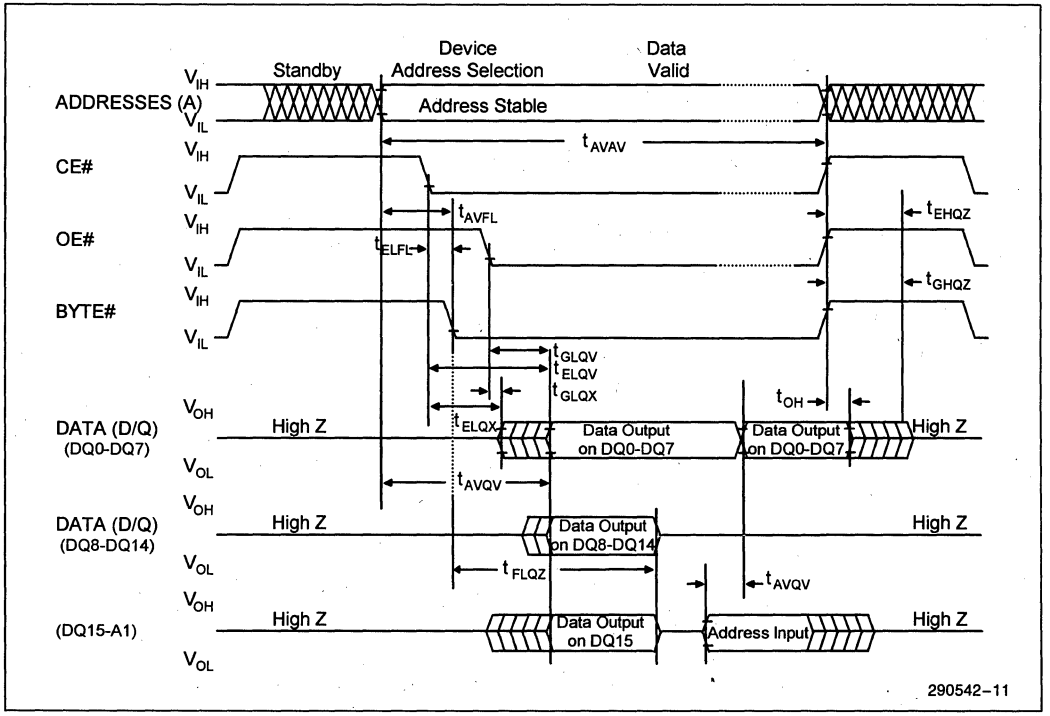


Figure 11. BYTE# Timing Diagram for Both Read and Write Operations with V_{CC} at 5V

Table 13. AC Characteristics: WE#—Controlled Write Operations[1] (Automotive Temperature)

Symbol	Parameter	Notes	Min	Max	Unit
t_{AVAV}	Write Cycle Time		80		ns
t_{PHWL}	RP# High Recovery to WE# Going Low		220		ns
t_{ELWL}	CE# Setup to WE# Going Low		0		ns
t_{PHHWH}	Boot Block Lock Setup to WE# Going High	6,8	100		ns
t_{VPWH}	V_{PP} Setup to WE# Going High	5,8	100		ns
t_{AVWH}	Address Setup to WE# Going High	3	60		ns
t_{DVWH}	Data Setup to WE# Going High	4	60		ns
t_{WLWH}	WE# Pulse Width		60		ns
t_{WHDX}	Data Hold Time from WE# High	4	0		ns
t_{WHAX}	Address Hold Time from WE# High	3	10		ns
t_{WHEH}	CE# Hold Time from WE# High		10		ns
t_{WHWL}	WE# Pulse Width High		20		ns
t_{WHQV1}	Duration of Word/Byte Write Operation	2,5	7		µs
t_{WHQV2}	Duration of Erase Operation (Boot)	2,5,6	0.4		s
t_{WHQV3}	Duration of Erase Operation (Parameter)	2,5	0.4		s
t_{WHQV4}	Duration of Erase Operation (Main)	2,5	0.7		s
t_{QWL}	V_{PP} Hold from Valid SRD	5,8	0		ns
t_{QVPH}	RP# V_{HH} Hold from Valid SRD	6,8	0		ns
t_{PHBR}	Boot-Block Relock Delay	7,8		100	ns

NOTES:
1. Read timing characteristics during write and erase operations are the same as during read-only operations. Refer to AC characteristics during read mode.
2. The on-chip WSM completely automates Program/Erase operations; Program/Erase algorithms are now controlled internally which includes verify and margining operations.
3. Refer to command definition table for valid A_{IN}.
4. Refer to command definition table for valid D_{IN}.
5. Program/Erase durations are measured to valid SRD data (successful operation, SR.7 = 1).
6. For boot block Program/Erase, RP# should be held at V_{HH} or WP# should be held at V_{IH} until operation completes successfully.
7. Time t_{PHBR} is required for successful relocking of the boot block.
8. Sampled, but not 100% tested.
9. V_{PP} at 5.0V.
10. V_{PP} at 12.0V.
11. See 5V Standard Test Configuration.

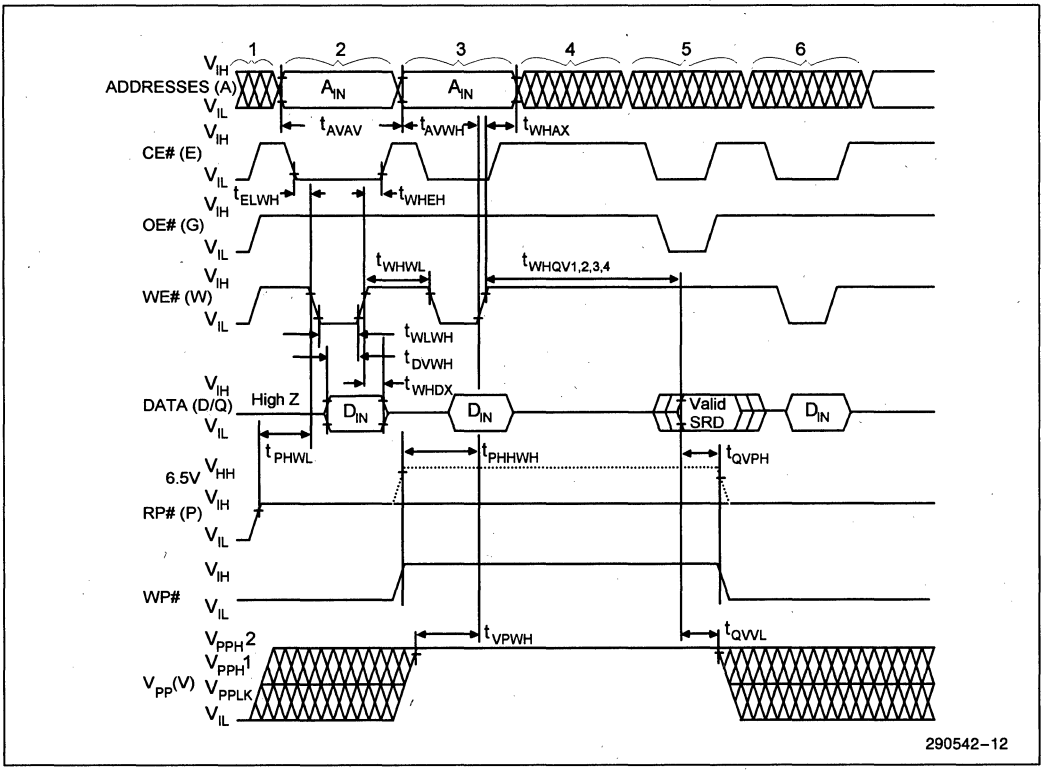


Figure 12. AC Waveforms for Write and Erase Operations (WE#-Controlled Writes)

Table 14. AC Characteristics: CE#-Controlled Write Operations(1,12)

Symbol	Parameter	Notes	Min	Max	Unit
t_{AVAV}	Write Cycle Time		80		ns
t_{PHEL}	RP# High Recovery to CE# Going Low			300	ns
t_{WLEL}	WE# Setup to CE# Going Low		0		ns
t_{PHHEH}	Boot Block Lock Setup to CE# Going High	6,8	100		ns
t_{VPEH}	V_{PP} Setup to CE# Going High	5,8	100		ns
t_{AVEH}	Address Setup to CE# Going High		60		ns
t_{DVEH}	Data Setup to CE# Going High	3	60		ns
t_{ELEH}	CE# Pulse Width	4	60		ns
t_{EHDX}	Data Hold Time from CE# High		0		ns
t_{EHAX}	Address Hold Time from CE# High	4	10		ns
t_{EHWH}	WE # Hold Time from CE# High	3	10		ns
t_{EHEL}	CE# Pulse Width High		20		ns
t_{EHQV1}	Duration of Word/Byte Write Operation	2,5	7		μs
t_{EHQV2}	Duration of Erase Operation (Boot)	2,5,6	0.4		s
t_{EHQV3}	Duration of Erase Operation (Parameter)	2,5	0.4		s
t_{EHQV4}	Duration of Erase Operation (Main)	2,5	0.7		s
t_{QWL}	V_{PP} Hold from Valid SRD	5,8	0		ns
t_{QVPH}	RP# V_{HH} Hold from Valid SRD	6,8	0		ns
t_{PHBR}	Boot-Block Relock Delay	7,8		100	ns

NOTES:
See WE# Controlled Write Operations for notes 1 through 11.

12. Chip-Enable controlled writes: Write operations are driven by the valid combination of CE# and WE# in systems where CE# defines the write pulse-width (within a longer WE# timing waveform), all set-up, hold and inactive WE# times should be measured relative to the CE# waveform.

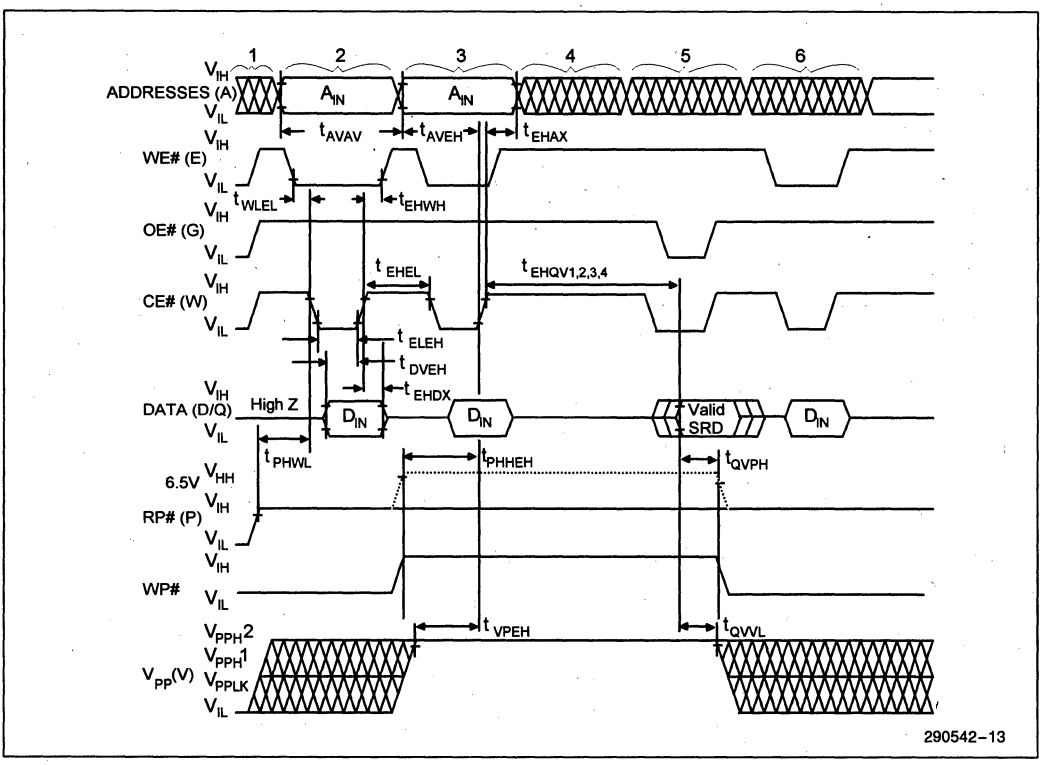


Figure 13. Alternate AC Waveforms for Write and Erase Operations (CE#-Controlled Writes)

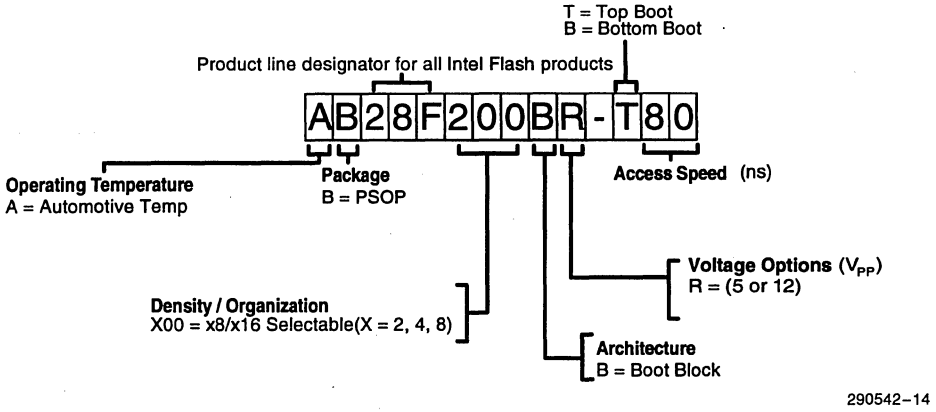
Table 15. Extended Temperature Operations—Erase and Program Timings

Parameter	V_{PP} = 5V ±10%		V_{PP} = 12V ±5%		Unit
	Typ	Max	Typ	Max	
Boot/Parameter Block Erase Time	0.8	7.8	0.34	4.0	s
Main Block Erase Time	1.9	15.4	1.1	7.1	s
Main Block Write Time (Byte Mode)	1.4	16.8	1.2	6.8	s
Main Block Write Time (Word Mode)	0.9	8.4	0.6	3.4	s

NOTE:
All numbers are sampled, not 100% tested.

6.0 ADDITIONAL INFORMATION

6.1 Ordering Information



VALID COMBINATIONS:
AB28F200BR-T80 AB28F200BR-B80

6.2 References

Order Number	Document
292130	AB-57 "Boot Block Architecture for Safe Firmware Updates"
292098	AP-363 "Extended Flash BIOS Concepts for Portable Computers"
290448	28F002/200BX-T/B 2-Mbit Boot Block Flash Memory Datasheet
290449	28F002/200BL-T/B 2-Mbit Low Power Boot Block Flash Memory Datasheet
290450	28F004/400BL-T/B 4-Mbit Low Power Boot Block Flash Memory Datasheet
290451	28F004/400BX-T/B 4-Mbit Boot Block Flash Memory Datasheet
290531	2-Mbit SmartVoltage Boot Block Flash Memory Family Datasheet

6.3 Revision History

Number	Description
-001	Initial release of datasheet

A28F200BX-T/B
2-MBIT (128K x 16, 256K x 8) BOOT BLOCK FLASH MEMORY FAMILY

Automotive

- x8/x16 Input/Output Architecture
  - A28F200BX-T, A28F200BX-B
  - For High Performance and High Integration 16-bit and 32-bit CPUs
- Optimized High Density Blocked Architecture
  - One 16 KB Protected Boot Block
  - Two 8 KB Parameter Blocks
  - One 96 KB Main Block
  - One 128 KB Main Block
  - Top or Bottom Boot Locations
- Extended Cycling Capability
  - 1,000 Block Erase Cycles
- Automated Word/Byte Write and Block Erase
  - Command User Interface
  - Status Register
  - Erase Suspend Capability
- SRAM-Compatible Write Interface
- Automatic Power Savings Feature
  - 1 mA Typical I_{CC} Active Current in Static Operation
- Hardware Data Protection Feature
  - Erase/Write Lockout during Power Transitions
- Very High-Performance Read
  - 90 ns Maximum Access Time
  - 45 ns Maximum Output Enable Time
- Low Power Consumption
  - 25 mA Typical Active Read Current
- Deep Power-Down/Reset Input
  - Acts as Reset for Boot Operations
- Automotive Temperature Operation
  - $-40°C$ to $+125°C$
- Write Protection for Boot Block
- Industry Standard Surface Mount Packaging
  - JEDEC ROM Compatible 44-Lead PSOP
- 12V Word/Byte Write and Block Erase
  - $V_{PP} = 12V \pm 5\%$ Standard
- ETOX™ III Flash Technology
  - 5V Read
- Independent Software Vendor Support

November 1994
Order Number: 290500-002

Intel's 2-Mbit Flash Memory Family is an extension of the Boot Block Architecture which includes block-selective erasure, automated write and erase operations and standard microprocessor interface. The 2 Mbit Flash Memory Family enhances the Boot Block Architecture by adding more density and blocks, x8/x16 input/output control, very high speed, low power, an industry standard ROM compatible pinout. The 2-Mbit flash family allows for an easy upgrade to Intel's 4-Mbit Boot Block Flash Memory Family.

The Intel A28F200BX-T/B are 16-bit wide flash memory offerings optimized to meet the rigorous environmental requirements of Automotive Applications. These high density flash memories provide user selectable bus operation for either 8-bit or 16-bit applications. The A28F200BX-T and A28F200BX-B are 2,097,152-bit nonvolatile memories organized as either 262,144 bytes or 131,072 words of information. They are offered in 44-Lead plastic SOP packages. The x8/x16 pinout conforms to the industry standard ROM/EPROM pinout. Read and Write Characteristics are guaranteed over the ambient temperature range of −40°C to +125°C.

These devices use an integrated Command User Interface (CUI) and Write State Machine (WSM) for simplified word/byte write and block erasure. The A28F200BX-T provides block locations compatible with Intel's MCS-186 family, 80286, i386™, i486™, i860™ and 80960CA microprocessors. The A28F200BX-B provides compatibility with Intel's 80960KX and 80960SX families as well as other embedded microprocessors.

The boot block includes a data protection feature to protect the boot code in critical applications. With a maximum access time of 90 ns, these 2 Mbit flash devices are very high performance memories which interface at zero-wait-state to a wide range of microprocessors and microcontrollers.

Manufactured on Intel's 0.8 micron ETOX™ III process, the 2-Mbit flash memory family provides world class quality, reliability and cost-effectiveness at the 2-Mbit density level.

1.0 PRODUCT FAMILY OVERVIEW

Throughout this datasheet the A28F200BX refers to both the A28F200BX-T and A28F200BX-B devices. Section 1 provides an overview of the 2-Mbit flash memory family including applications, pinouts and pin descriptions. Section 2 describes in detail the specific memory organization. Section 3 provides a description of the family's principles of operation. Finally, the family's operating specifications are described.

1.1 Main Features

The A28F200BX boot block flash memory family is a very high performance 2-Mbit (2,097,152 bit) memory family organized as either 128-KWords (131,072 words) of 16 bits each or 256-Kbytes (262,144 bytes) of 8 bits each.

Five Separately Erasable Blocks including a **hardware-lockable boot block** (16,384 Bytes), **two parameter blocks** (8,192 Bytes each) and **two main blocks** (1 block of 98,304 Bytes and 1 block of 131,072 Bytes) are included on the 2-Mbit family. An erase operation erases one of the main blocks in typically 3 seconds, and the boot or parameter blocks in typically 1.5 seconds. Each block can be independently erased and programmed 1,000 times.

The Boot Block is located at either the top (A28F200BX-T) or the bottom (A28F200BX-B) of the address map in order to accommodate different microprocessor protocols for boot code location. The **hardware lockable boot block** provides the most secure code storage. The boot block is intended to store the kernel code required for booting-up a system. When the **RP#** pin is between 11.4V and 12.6V the boot block is unlocked and program and erase operations can be performed. When the RP# pin is at or below 6.5V the boot block is locked and program and erase operations to the boot block are ignored.

The A28F200BX products are available in the ROM/EPROM compatible pinout and housed in the 44-Lead PSOP (Plastic Small Outline) package as shown in Figure 3.

The **Command User Interface (CUI)** serves as the interface between the microprocessor or microcontroller and the internal operation of the A28F200BX flash memory.

Program and Erase Automation allows program and erase operations to be executed using a two-write command sequence to the CUI. The internal Write State Machine (WSM) automatically executes the algorithms and timings necessary for program and erase operations, including verifications, thereby unburdening the microprocessor or microcontroller. Writing of memory data is performed in word or byte increments for the A28F200BX family typically within 9 μs which is a 100% improvement over previous flash memory products.

The **Status Register (SR)** indicates the status of the WSM and whether the WSM successfully completed the desired program or erase operation.

Maximum Access Time of **90 ns (TACC)** is achieved over the automotive temperature range (−40°C to 125°C), 10% V_{CC} supply voltage range and 100 pF output load.

I_{PP} maximum Program current is 40 mA for x16 operation and 30 mA for x8 operation. I_{PP} Erase current is 30 mA maximum. V_{PP} erase and programming voltage is 11.4V to 12.6V (V_{PP} = 12V ±5%) under all operating conditions. Typical I_{CC} Active Current of 25 mA is achieved.

The 2-Mbit boot block flash family is also designed with an Automatic Power Savings (APS) feature to minimize system battery current drain and allow for very low power designs. Once the device is accessed to read array data, APS mode will immediately put the memory in static mode of operation where I_{CC} active current is typically 1 mA until the next read is initiated.

When the CE# and RP# pins are at V_{CC} and the BYTE# pin is at either V_{CC} or GND the **CMOS Standby** mode is enabled where I_{CC} is typically 80 μA.

A **Deep Power-Down Mode** is enabled when the RP# pin is at ground minimizing power consumption and providing write protection during power-up conditions. **I_{CC} current** during deep power-down mode is **50 μA typical**. An initial maximum access time or Reset Time of 300 ns is required from RP# switching until outputs are valid. Equivalently, the device has a maximum wake-up time of 210 ns until writes to the Command User Interface are recognized.

When RP# is at ground the WSM is reset, the Status Register is cleared and the entire device is protected from being written to. This feature prevents data corruption and protects the code stored in the device during system reset. The system Reset pin can be tied to RP# to reset the memory to normal read mode upon activation of the Reset pin. With on-chip program/erase automation in the 2 Mbit family and the RP# functionality for data protection, when the CPU is reset and even if a program or erase command is issued, the device will not recognize any operation until RP# returns to its normal state.

For the A28F200BX, Byte-wide or Word-wide Input/Output Control is possible by controlling the BYTE# pin. When the BYTE# pin is at a logic low the device is in the byte-wide mode (x8) and data is read and written through DQ[0:7]. During the byte-wide mode, DQ[8:14] are tri-stated and DQ15/A−1 becomes the lowest order address pin. When the BYTE# pin is at a logic high the device is in the word-wide mode (x16) and data is read and written through DQ[0:15].

1.2 Applications

The 2-Mbit boot block flash family combines high density, high performance, cost-effective flash memories with blocking and hardware protection capabilities. Its flexibility and versatility will reduce costs throughout the product life cycle. Flash memory is ideal for Just-In-Time production flow, reducing system inventory and costs, and eliminating component handling during the production phase. During the product life cycle, when code updates or feature enhancements become necessary, flash memory will reduce the update costs by allowing either a user-performed code change via floppy disk or a remote code change via a serial link. The 2-Mbit boot block flash family provides full function, blocked flash memories suitable for a wide range of automotive applications.

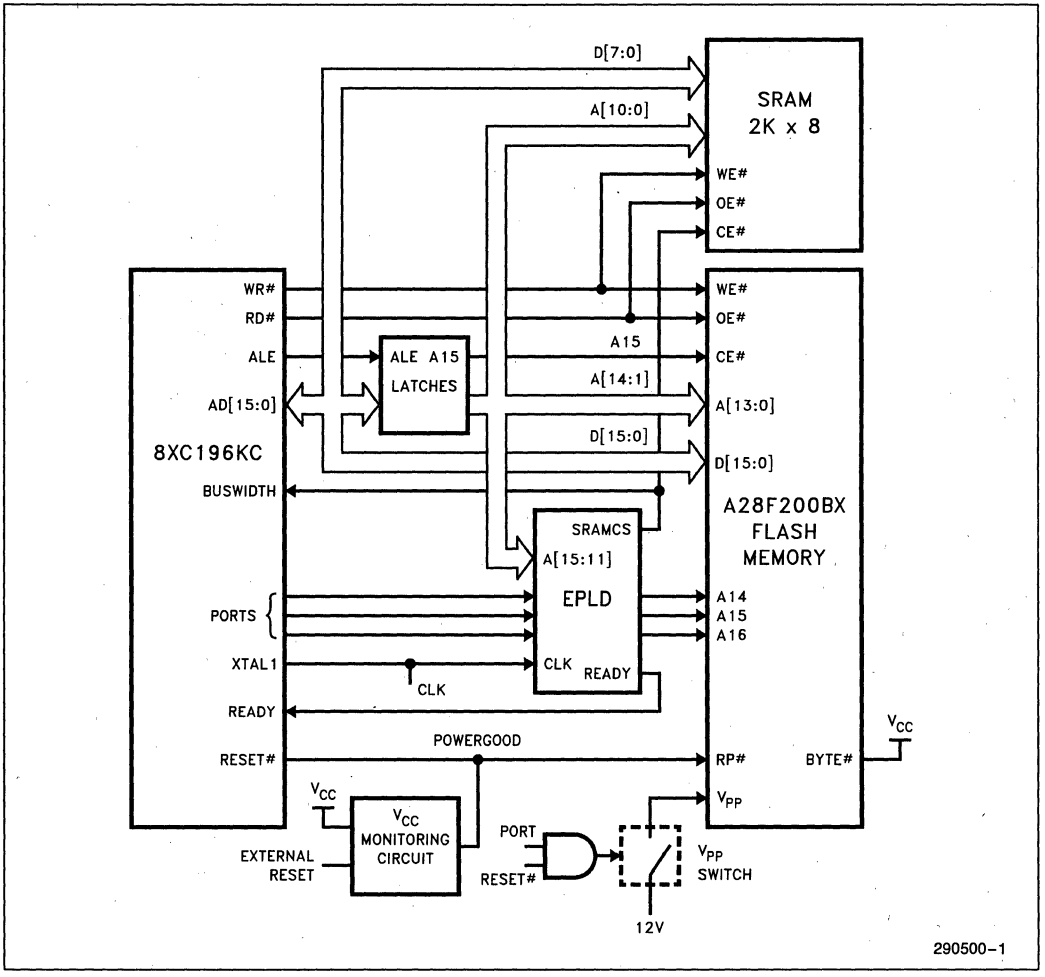


Figure 1. A28F200BX Interface to 8XC196KC

1.3 Pinouts

The A28F200BX 44-Lead PSOP pinout follows the industry standard ROM/EPROM pinout as shown in Figure 2 with an upgrade to the 28F400BC (4-Mbit flash family).

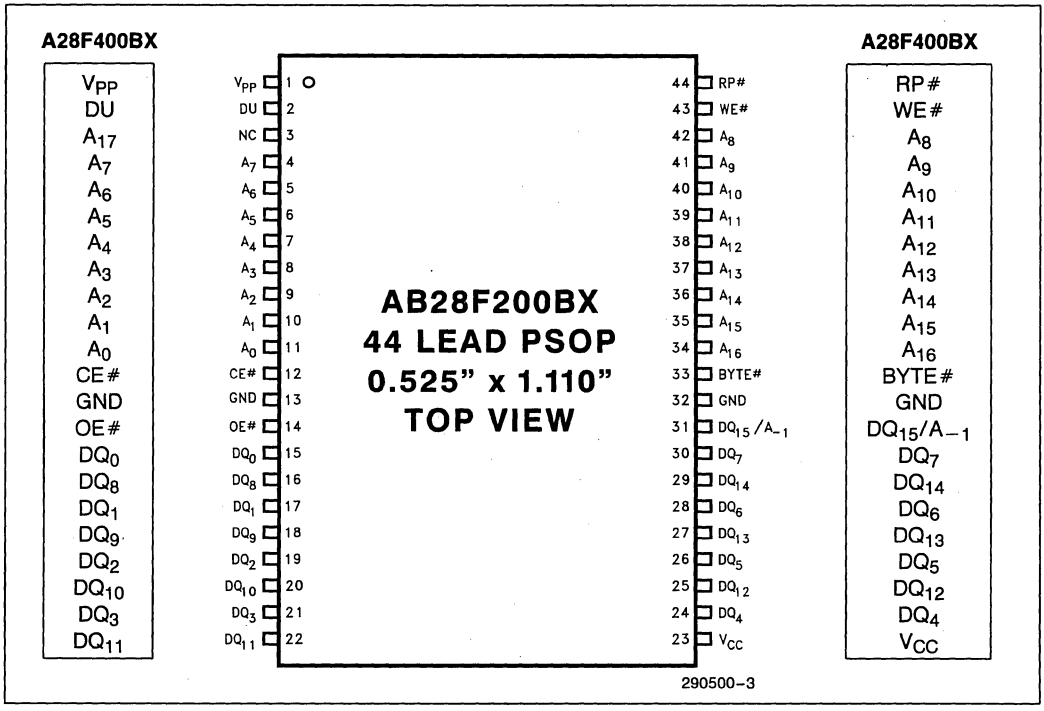


Figure 2. PSOP Lead Configuration

8

1.4 Pin Descriptions for the x8/x16 A28F200BX

Symbol	Type	Name and Function
A_0–A_{16}	I	**ADDRESS INPUTS** for memory addresses. Addresses are internally latched during a write cycle.
A_9	I	**ADDRESS INPUT:** When A_9 is at 12V the signature mode is accessed. During this mode A_0 decodes between the manufacturer and device ID's. When BYTE# is at a logic low only the lower byte of the signatures are read. DQ_{15}/A_{-1} is a don't care in the signature mode when BYTE# is low.
DQ_0–DQ_7	I/O	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a program command. Inputs commands to the Command User Interface when CE# and WE# are active. Data is internally latched during the write and program cycles. Outputs array, intelligent identifier and Status Register data. The data pins float to tri-state when the chip is deselected or the outputs are disabled.
DQ_8–DQ_{15}	I/O	**DATA INPUTS/OUTPUTS:** Inputs array data on the second CE# and WE# cycle during a program command. Data is internally latched during the write and program cycles. Outputs array data. The data pins float to tri-state when the chip is deselected or the outputs are disabled as in the byte-wide mode (BYTE# = "0"). In the byte-wide mode DQ_{15}/A_{-1} becomes the lowest order address for data output on DQ_0–DQ_7.
CE#	I	**CHIP ENABLE:** Activates the device's control logic, input buffers, decoders and sense amplifiers. CE# is active low; CE# high deselects the memory device and reduces power consumption to standby levels. If CE# and RP# are high, but not at a CMOS high level, the standby current will increase due to current flow through the CE# and RP# input stages.
RP#	I	**RESET/POWER-DOWN:** Provides three-state control. Puts the device in deep power-down mode. Locks the boot block from program/erase. When RP# is at logic high level and equals 6.5V maximum the boot block is locked and cannot be programmed or erased. When RP# = 11.4V minimum the boot block is unlocked and can be programmed or erased. When RP# is at a logic low level the boot block is locked, the deep power-down mode is enabled and the WSM is reset preventing any blocks from being programmed or erased, therefore providing data protection during power transitions. When RP# transitions from logic low to logic high the flash memory enters the read array mode.
OE#	I	**OUTPUT ENABLE:** Gates the device's outputs through the data buffers during a read cycle. OE# is active low.
WE#	I	**WRITE ENABLE:** Controls writes to the Command Register and array blocks. WE# is active low. Addresses and data are latched on the rising edge of the WE# pulse.
BYTE#	I	**BYTE# ENABLE:** Controls whether the device operates in the byte-wide mode (x8) or the word-wide mode (x16). BYTE# pin must be controlled at CMOS levels to meet 130 μA CMOS current in the standby mode. BYTE# = "0" enables the byte-wide mode, where data is read and programmed on DQ_0–DQ_7 and DQ_{15}/A_{-1} becomes the lowest order address that decodes between the upper and lower byte. DQ_8–DQ_{14} are tri-stated during the byte-wide mode. BYTE# = "1" enables the word-wide mode where data is read and programmed on DQ_0–DQ_{15}.
V_{PP}		**PROGRAM/ERASE POWER SUPPLY:** For erasing memory array blocks or programming data in each block. **Note:** $V_{PP} < V_{PPLMAX}$ memory contents cannot be altered.
V_{CC}		**DEVICE POWER SUPPLY (5V ± 10%)**
GND		**GROUND:** For all internal circuitry.
NC		**NO CONNECT:** Pin may be driven or left floating.
DU		**DON'T USE PIN:** Pin should not be connected to anything.

2.0 A28F200BX WORD/BYTE-WIDE PRODUCTS DESCRIPTION

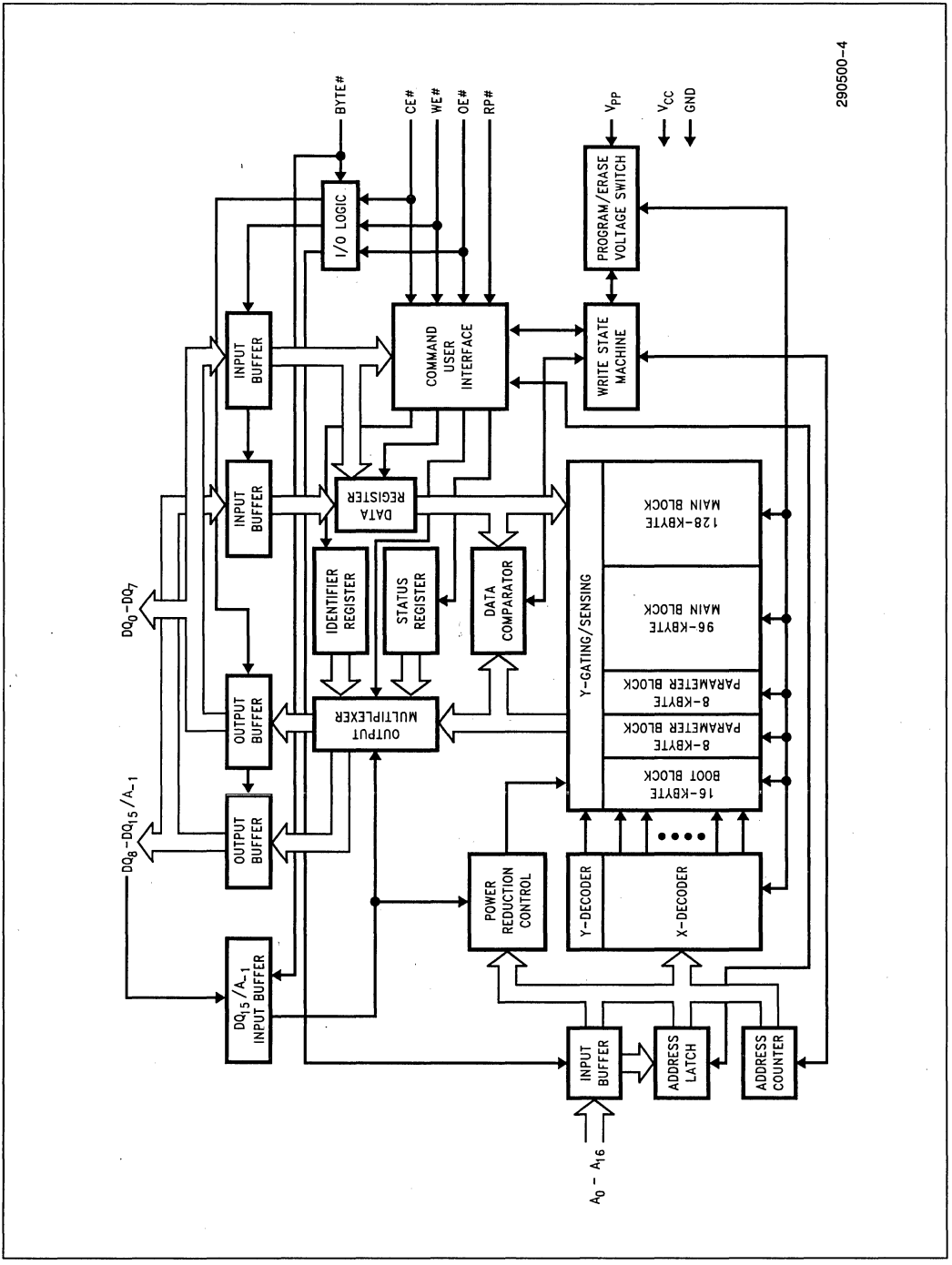


Figure 3. A28F200BX Word/Byte-Wide Block Diagram

8

2.1 A28F200BX Memory Organization

2.1.1 BLOCKING

The A28F200BX uses a blocked array architecture to provide independent erasure of memory blocks. A block is erased independently of other blocks in the array when an address is given within the block address range and the Erase Setup and Erase Confirm commands are written to the CUI. The A28F200BX is a random read/write memory, only erasure is performed by block.

2.1.1.1 Boot Block Operation and Data Protection

The 16-Kbyte boot block provides a lock feature for secure code storage. The intent of the boot block is to provide a secure storage area for the kernel code that is required to boot a system in the event of power failure or other disruption during code update. This lock feature ensures absolute data integrity by preventing the boot block from being written or erased when RP# is not at 12V. The boot block can be erased and written when RP# is held at 12V for the duration of the erase or program operation. This allows customers to change the boot code when necessary while providing security when needed. See the Block Memory Map section for address locations of the boot block for the A28F200BX-T and A28F200BX-B.

2.1.1.2 Parameter Block Operation

The A28F200BX has 2 parameter blocks (8-Kbytes each). The parameter blocks are intended to provide storage for frequently updated system parameters and configuration or diagnostic information. The parameter blocks can also be used to store additional boot or main code. The parameter blocks however, do not have the hardware write protection feature that the boot block has. The parameter blocks provide for more efficient memory utilization when dealing with parameter changes versus regularly blocked devices. See the Block Memory Map section for address locations of the parameter blocks for the A28F200BX-T and A28F200BX-B.

2.1.1.3 Main Block Operation

Two main blocks of memory exist on the A28F200BX (1 x 128-Kbyte block and 1 x 96-Kbyte block). See the following section on Block Memory Map for the address location of these blocks for the A28F200BX-T

and A28F200BX-B products.

2.1.2 BLOCK MEMORY MAP

Two versions of the A28F200BX product exist to support two different memory maps of the array blocks in order to accommodate different microprocessor protocols for boot code location. The A28F200BX-T memory map is inverted from the A28F200BX-B memory map.

2.1.2.1 A28F200BX-B Memory Map

The A28F200BX-B device has the 16-Kbyte boot block located from 00000H to 01FFFH to accommodate those microprocessors that boot from the bottom of the address map at 00000H. In the A28F200BX-B the first 8-Kbyte parameter block resides in memory space from 02000H to 02FFFH. The second 8-Kbyte parameter block resides in memory space from 03000H to 03FFFH. The 96-Kbyte main block resides in memory space from 04000H to 0FFFFH. The 128-Kbyte main block resides in memory space from 10000H to 1FFFFH (word locations). See Figure 4.

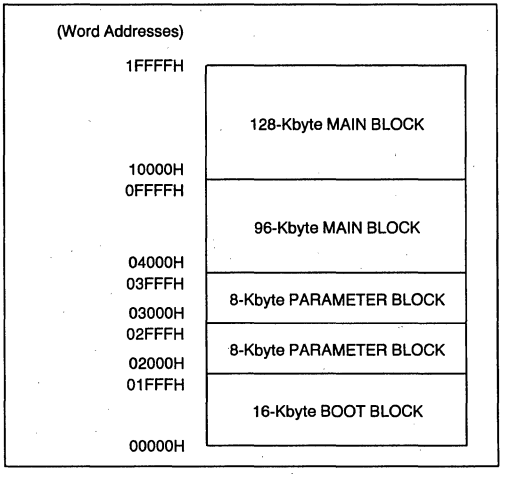


Figure 4. A28F200BX-B Memory Map

2.1.2.2 A28F200BX-T Memory Map

The A28F200BX-T device has the 16-Kbyte boot block located from 1E000H to 1FFFFH to accommodate those microprocessors that boot from the top of the address map. In the A28F200BX-T the first 8 Kbyte parameter block resides in memory space from 1D000H to 1DFFFH. The second 8-Kbyte parameter block resides in memory space from 1C000H to 1CFFFH. The 96-Kbyte main block resides in memory space from 10000H to 1BFFFH. The 128-Kbyte main block resides in memory space from 00000H to 0FFFFH as shown in Figure 5.

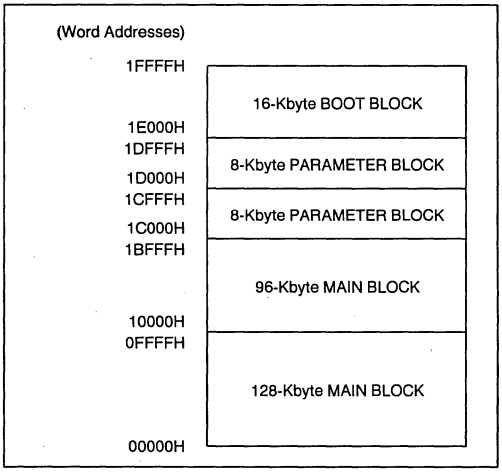


Figure 5. A28F200BX-T Memory Map

3.0 PRODUCT FAMILY PRINCIPLES OF OPERATION

Flash memory augments EPROM functionality with in-circuit electrical write and erase. The 2-Mbit flash family utilizes a Command User Interface (CUI) and internally generated and timed algorithms to simplify write and erase operations.

The CUI allows for 100% TTL-level control inputs, fixed power supplies during erasure and programming, and maximum EPROM compatibility.

In the absence of high voltage on the V_{PP} pin, the 2-Mbit boot block flash family will only successfully execute the following commands: Read Array, Read Status Register, Clear Status Register and Intelligent Identifier mode. The device provides standard EPROM read, standby and output disable operations. Manufacturer Identification and Device Identification data can be accessed through the CUI or through the standard EPROM A9 high voltage access (V_{ID}) for PROM programming equipment.

The same EPROM read, standby and output disable functions are available when high voltage is applied to the V_{PP} pin. In addition, high voltage on V_{PP} allows write and erase of the device. All functions associated with altering memory contents: write and erase, Intelligent Identifier read and Read Status are accessed via the CUI.

The purpose of the Write State Machine (WSM) is to completely automate the write and erasure of the device. The WSM will begin operation upon receipt of a signal from the CUI and will report status back through a Status Register. The CUI will handle the WE# interface to the data and address latches, as well as system software requests for status while the WSM is in operation.

3.1 Bus Operations

Flash memory reads, erases and writes in-system via the local CPU. All bus cycles to or from the flash memory conform to standard microprocessor bus cycles.

8

Table 1. Bus Operations for WORD-WIDE Mode (BYTE# = V_{IH})

Mode	Notes	RP#	CE#	OE#	WE#	A_9	A_0	V_{PP}	DQ_{0-15}
Read	1, 2, 3	V_{IH}	V_{IL}	V_{IL}	V_{IH}	X	X	X	D_{OUT}
Output Disable		V_{IH}	V_{IL}	V_{IH}	V_{IH}	X	X	X	High Z
Standby		V_{IH}	V_{IH}	X	X	X	X	X	High Z
Deep Power-Down	9	V_{IL}	X	X	X	X	X	X	High Z
Intelligent Identifier (Mfr)	4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IL}	X	0089H
Intelligent Identifier (Device)	4, 5	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IH}	X	2274H 2275H
Write	6, 7, 8	V_{IH}	V_{IL}	V_{IH}	V_{IL}	X	X	X	D_{IN}

Table 2. Bus Operations for BYTE-WIDE Mode (BYTE# = V_{IL})

Mode	Notes	RP#	CE#	OE#	WE#	A_9	A_0	A_{-1}	V_{PP}	DQ_{0-7}	DQ_{8-14}
Read	1, 2, 3	V_{IH}	V_{IL}	V_{IL}	V_{IH}	X	X	X	X	D_{OUT}	High Z
Output Disable		V_{IH}	V_{IL}	V_{IH}	V_{IH}	X	X	X	X	High Z	High Z
Standby		V_{IH}	V_{IH}	X	X	X	X	X	X	High Z	High Z
Deep Power-Down	9	V_{IL}	X	X	X	X	X	X	X	High Z	High Z
Intelligent Identifier (Mfr)	4	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IL}	X	X	89H	High Z
Intelligent Identifier (Device)	4, 5	V_{IH}	V_{IL}	V_{IL}	V_{IH}	V_{ID}	V_{IH}	X	X	74H 75H	High Z
Write	6, 7, 8	V_{IH}	V_{IL}	V_{IH}	V_{IL}	X	X	X	X	D_{IN}	High Z

NOTES:
1. Refer to DC Characteristics.
2. X can be V_{IL} or V_{IH} for control pins and addresses, V_{PPL} or V_{PPH} for V_{PP}.
3. See DC characteristics for V_{PPL}, V_{PPH}, V_{HH}, V_{ID} voltages.
4. Manufacturer and Device codes may also be accessed via a CUI write sequence. A_1–A_{17} = X.
5. Device ID = 2274H for A28F200BX-T and 2275H for A28F200BX-B.
6. Refer to Table 3 for valid D_{IN} during a write operation.
7. Command writes for Block Erase or Word/Byte Write are only executed when V_{PP} = V_{PPH}.
8. To write or erase the boot block, hold RP# at V_{HH}.
9. RP# must be at GND ±0.2V to meet the 80 µA maximum deep power-down current.

3.2 Read Operations

The 2-Mbit boot block flash family has three user read modes; Array, Intelligent Identifier, and Status Register. Status Register read mode will be discussed in detail in the "Write Operations" section.

During power-up conditions (V_{CC} supply ramping), it takes a maximum of 300 ns from when V_{CC} is at 4.5V minimum to valid data on the outputs.

3.2.1 READ ARRAY

If the memory is not in the Read Array mode, it is necessary to write the appropriate read mode command to the CUI. The 2-Mbit boot block flash family has three control functions, all of which must be logically active, to obtain data at the outputs. Chip-Enable CE# is the device selection control. Reset/Power-Down RP# is the device power control. Output-Enable OE# is the DATA INPUT/OUTPUT (DQ[0:15] or DQ[0:7]) direction control and when active is used to drive data from the selected memory on to the I/O bus.

3.2.1.1 Output Control

With OE# at logic-high level (V_{IH}), the output from the device is disabled and data input/output pins (DQ[0:15] or DQ[0:7]) are tri-stated. Data input is then controlled by WE#.

3.2.1.2 Input Control

With WE# at logic-high level (V_{IH}), input to the device is disabled. Data Input/Output pins (DQ-[0:15] or DQ[0:7]) are controlled by OE#.

3.2.2 INTELLIGENT IDENTIFIERS

The manufacturer and device codes are read via the CUI or by taking the A_9 pin to 12V. Writing 90*H* to the CUI places the device into Intelligent Identifier read mode. A read of location 00000*H* outputs the manufacturer's identification code, 0089H, and location 00001*H* outputs the device code; 2274H for A28F200BX-T, 2275H for A28F200BX-B. When BYTE# is at a logic low only the lower byte of the above signatures is read and DQ_{15}/A_{-1} is a "don't care" during Intelligent Identifier mode. A read array command must be written to the CUI to return to the read array mode.

3.3 Write Operations

Commands are written to the CUI using standard microprocessor write timings. The CUI serves as the interface between the microprocessor and the internal chip operation. The CUI can decipher Read Array, Read Intelligent Identifier, Read Status Register, Clear Status Register, Erase and Program commands. In the event of a read command, the CUI simply points the read path at either the array, the Intelligent Identifier, or the status register depending on the specific read command given. For a program or erase cycle, the CUI informs the write state machine that a write or erase has been requested. During a program cycle, the Write State Machine will control the program sequences and the CUI will only respond to status reads. During an erase cycle, the CUI will respond to status reads and erase suspend. After the Write State Machine has completed its task, it will allow the CUI to respond to its full command set. The CUI will stay in the current command state until the microprocessor issues another command.

The CUI will successfully initiate an erase or write operation only when V_{PP} is within its voltage range. Depending upon the application, the system designer may choose to make the V_{PP} power supply switchable, available only when memory updates are desired. The system designer can also choose to "hard-wire" V_{PP} to 12V. The 2-Mbit boot block flash family is designed to accommodate either design practice. It is strongly recommended that RP# be tied to logical Reset for data protection during unstable CPU reset function as described in the "Product Family Overview" section.

3.3.1 BOOT BLOCK WRITE OPERATIONS

In the case of Boot Block modifications (write and erase), RP# is set to V_{HH} = 12V typically, in addition to V_{PP} at high voltage. However, if RP# is not at V_{HH} when a program or erase operation of the boot block is attempted, the corresponding status register bit (Bit 4 for Program and Bit 5 for Erase, refer to Table 4 for Status Register Definitions) is set to indicate the failure to complete the operation.

3.3.2 COMMAND USER INTERFACE (CUI)

The Command User Interface (CUI) serves as the interface to the microprocessor. The CUI points the read/write path to the appropriate circuit block as described in the previous section. After the WSM has completed its task, it will set the WSM Status bit to a "1", which will also allow the CUI to respond to its full command set. Note that after the WSM has returned control to the CUI, the CUI will remain in its current state.

3.3.2.1 Command Set

Command Codes	Device Mode
00	Invalid/Reserved
10	Alternate Program Setup
20	Erase Setup
40	Program Setup
50	Clear Status Register
70	Read Status Register
90	Intelligent Identifier
B0	Erase Suspend
D0	Erase Resume/Erase Confirm
FF	Read Array

3.3.2.2 Command Function Descriptions

Device operations are selected by writing specific commands into the CUI. Table 3 defines the 2-Mbit boot block flash family commands.

Table 3. Command Definitions

Command	Bus Cycles Req'd	Notes	First Bus Cycle			Second Bus Cycle		
		8	Operation	Address	Data	Operation	Address	Data
Read Array/Reset	1	1	Write	X	FFH			
Intelligent Identifier	3	2, 4	Write	X	90H	Read	IA	IID
Read Status Register	2	3	Write	X	70H	Read	X	SRD
Clear Status Register	1		Write	X	50H			
Erase Setup/Erase Confirm	2	5	Write	BA	20H	Write	BA	D0H
Word/Byte Write Setup/Write	2	6, 7	Write	WA	40H	Write	WA	WD
Erase Suspend/Erase Resume	2		Write	X	B0H	Write	X	D0H
Alternate Word/Byte Write Setup/Write	2	6, 7	Write	WA	10H	Write	WA	WD

NOTES:
1. Bus operations are defined in Tables 1, 2.
2. IA = Identifier Address: 00H for manufacturer code, 01H for device code.
3. SRD = Data read from Status Register.
4. IID = Intelligent Identifier Data.
Following the Intelligent Identifier Command, two read operations access manufacturer and device codes.
5. BA = Address within the block being erased.
6. PA = Address to be programmed.
PD = Data to be programmed at location PA.
7. Either 40H or 10H command is valid.
8. When writing commands to the device, the upper data bus [DQ8–DQ15] = X which is either V_{CC} or V_{SS} to avoid burning additional current.

Invalid/Reserved

These are unassigned commands. It is not recommended that the customer use any command other than the valid commands specified above. Intel reserves the right to redefine these codes for future functions.

Read Array (FFH)

This single write command points the read path at the array. If the host CPU performs a CE#/OE# controlled read immediately following a two-write sequence that started the WSM, then the device will output status register contents. If the Read Array command is given after Erase Setup the device is reset to read the array. A two Read Array command sequence (FFH) is required to reset to Read Array after Program Setup.

Intelligent Identifier (90H)

After this command is executed, the CUI points the output path to the Intelligent Identifier circuits. Only Intelligent Identifier values at addresses 0 and 1 can be read (only address A0 is used in this mode, all other address inputs are ignored).

Read Status Register (70H)

This is one of the two commands that is executable while the state machine is operating. After this command is written, a read of the device will output the contents of the status register, regardless of the address presented to the device.

The device automatically enters this mode after program or erase has completed.

Clear Status Register (50H)

The WSM can only set the Program Status and Erase Status bits in the status register, it can not clear them. Two reasons exist for operating the status register in this fashion. The first is a synchronization. The WSM does not know when the host CPU has read the status register, therefore it would not know when to clear the status bits. Secondly, if the CPU is programming a string of bytes, it may be more efficient to query the status register after programming the string. Thus, if any errors exist while programming the string, the status register will return the accumulated error status.

Program Setup (40H or 10H)

This command simply sets the CUI into a state such that the next write will load the address and data registers. Either 40H or 10H can be used for Program Setup. Both commands are included to accommodate efforts to achieve an industry standard command code set.

Program

The second write after the program setup command, will latch addresses and data. Also, the CUI initiates the WSM to begin execution of the program algorithm. While the WSM finishes the algorithm, the device will output Status Register contents. Note that the WSM cannot be suspended during programming.

Erase Setup (20H)

Prepares the CUI for the Erase Confirm command. No other action is taken. If the next command is not an Erase Confirm command then the CUI will set both the Program Status and Erase Status bits of the Status Register to a "1", place the device into the Read Status Register state, and wait for another command.

Erase Confirm (D0H)

If the previous command was an Erase Setup command, then the CUI will enable the WSM to erase, at the same time closing the address and data latches, and respond only to the Read Status Register and Erase Suspend commands. While the WSM is executing, the device will output Status Register data when OE# is toggled low. Status Register data can only be updated by toggling either OE# or CE# low.

Erase Suspend (B0H)

This command only has meaning while the WSM is executing an Erase operation, and therefore will only be responded to during an erase operation. After this command has been executed, the CUI will set an output that directs the WSM to suspend Erase operations, and then return to responding to only Read Status Register or to the Erase Resume commands. Once the WSM has reached the Suspend state, it will set an output into the CUI which allows the CUI to respond to the Read Array, Read Status Register, and Erase Resume commands. In this mode, the CUI will not respond to any other commands. The WSM will also set the WSM Status bit to a "1". The WSM will continue to run, idling in the SUSPEND state, regardless of the state of all input control pins, with the exclusion of RP#. RP# will immediately shut down the WSM and the remainder of the chip. During a suspend operation, the data and address latches will remain closed, but the address pads are able to drive the address into the read path.

Erase Resume (D0H)

This command will cause the CUI to clear the Suspend state and set the WSM Status bit to a "0", but only if an Erase Suspend command was previously issued. Erase Resume will not have any effect in all other conditions.

3.3.3 STATUS REGISTER

The 2-Mbit boot block flash family contains a status register which may be read to determine when a program or erase operation is complete, and whether that operation completed successfully. The status register may be read at any time by writing the Read Status command to the CUI. After writing this command, all subsequent Read operations output data from the status register until another command is written to the CUI. A Read Array command must be written to the CUI to return to the Read Array mode.

The status register bits are output on DQ[0:7] whether the device is in the byte-wide (x8) or word-wide (x16) mode. In the word-wide mode the upper byte, DQ[8:15] is set to 00H during a Read Status command. In the byte-wide mode, DQ[8:14] are tri-stated and DQ15/A−1 retains the low order address function.

It should be noted that the contents of the status register are latched on the falling edge of OE# or CE# whichever occurs last in the read cycle. This prevents possible bus errors which might occur if the contents of the status register change while reading the status register. CE# or OE# must be toggled with each subsequent status read, or the completion of a program or erase operation will not be evident.

The Status Register is the interface between the microprocessor and the Write State Machine (WSM). When the WSM is active, this register will indicate the status of the WSM, and will also hold the bits indicating whether or not the WSM was successful in performing the desired operation. The WSM sets status bits "Three" through "Seven" and clears bits "Six" and "Seven", but cannot clear status bits "Three" through "Five". These bits can only be cleared by the controlling CPU through the use of the Clear Status Register command.

3.3.3.1 Status Register Bit Definition

Table 4. Status Register Definitions

WSMS	ESS	ES	PS	VPPS	R	R	R
7	6	5	4	3	2	1	0

	NOTES:
SR.7 = WRITE STATE MACHINE STATUS 1 = Ready 0 = Busy	Write State Machine Status bit must first be checked to determine byte/word program or block erase completion, before the Program or Erase Status bits are checked for success.
SR.6 = ERASE SUSPEND STATUS 1 = Erase Suspended 0 = Erase in Progress/Completed	When Erase Suspend is issued, WSM halts execution and sets both WSMS and ESS bits to "1". ESS bit remains set to "1" until an Erase Resume command is issued.
SR.5 = ERASE STATUS 1 = Error in Block Erasure 0 = Successful Block Erase	When this bit is set to "1". WSM has applied the maximum number of erase pulses to the block and is still unable to successfully perform an erase verify.
SR.4 = PROGRAM STATUS 1 = Error in Byte/Word Program 0 = Successful Byte/Word Program	When this bit is set to "1", WSM has attempted but failed to Program a byte or word.
SR.3 = V_{PP} STATUS 1 = V_{PP} Low Detect; Operation Abort 0 = V_{PP} OK	The V_{PP} Status bit, unlike an A/D converter, does not provide continuous indication of V_{PP} level. The WSM interrogates the V_{PP} level only after the byte write or block erase command sequences have been entered and informs the system if V_{PP} has not been switched on. The V_{PP} Status bit is not guaranteed to report accurate feedback between V_{PPL} and V_{PPH}.
SR.2–SR.0 = RESERVED FOR FUTURE ENHANCEMENTS	These bits are reserved for future use and should be masked out when polling the Status Register.

3.3.3.2 Clearing the Status Register

Certain bits in the status register are set by the write state machine, and can only be reset by the system software. These bits can indicate various failure conditions. By allowing the system software to control the resetting of these bits, several operations may be performed (such as cumulatively programming several bytes or erasing multiple blocks in sequence). The status register may then be read to determine if an error occurred during that programming or erasure series. This adds flexibility to the way the device may be programmed or erased. To clear the status register, the Clear Status Register command is written to the CUI. Then, any other command may be issued to the CUI. Note again that before a read cycle can be initiated, a Read Array command must be written to the CUI to specify whether the read data is to come from the array, status register, or Intelligent Identifier.

3.3.4 PROGRAM MODE

Program is executed by a two-write sequence. The Program Setup command is written to the CUI followed by a second write which specifies the address and data to be programmed. The write state machine will execute a sequence of internally timed events to:

1. Program the desired bits of the addressed memory word (byte), and
2. Verify that the desired bits are sufficiently programmed

Programming of the memory results in specific bits within a byte or word being changed to a "0".

If the user attempts to program "1"s, there will be no change of the memory cell content and no error occurs.

Similar to erasure, the status register indicates whether programming is complete. While the program sequence is executing, bit 7 of the status register is a "0". The status register can be polled by

toggling either CE# or OE# to determine when the program sequence is complete. Only the Read Status Register command is valid while programming is active.

When programming is complete, the status bits, which indicate whether the program operation was successful, should be checked. If the programming operation was unsuccessful, Bit 4 of the status register is set to a "1" to indicate a Program Failure. If Bit 3 is set then V_{PP} was not within acceptable limits, and the WSM will not execute the programming sequence.

The status register should be cleared before attempting the next operation. Any CUI instruction can follow after programming is completed; however, it must be recognized that reads from the memory, status register, or Intelligent Identifier cannot be accomplished until the CUI is given the appropriate command. A Read Array command must first be given before memory contents can be read.

Figure 6 shows a system software flowchart for device byte programming operation. Figure 7 shows a similar flowchart for device word programming operation (A28F200BX-only).

3.3.5 ERASE MODE

Erasure of a single block is initiated by writing the Erase Setup and Erase Confirm commands to the CUI, along with the addresses, A[12:16], identifying the block to be erased. These addresses are latched internally when the Erase Confirm command is issued. Block erasure results in all bits within the block being set to "1".

The WSM will execute a sequence of internally timed events to:

1. Program all bits within the block
2. Verify that all bits within the block are sufficiently programmed
3. Erase all bits within the block and
4. Verify that all bits within the block are sufficiently erased

While the erase sequence is executing, Bit 7 of the status register is a "0".

When the status register indicates that erasure is complete, the status bits, which indicate whether the erase operation was successful, should be checked. If the erasure operation was unsuccessful, Bit 5 of the status register is set to a "1" to indicate an Erase Failure. If V_{PP} was not within acceptable limits after the Erase Confirm command is issued, the WSM will not execute an erase sequence; instead, Bit 5 of the status register is set to a "1" to indicate an Erase Failure, and Bit 3 is set to a "1" to identify that V_{PP} supply voltage was not within acceptable limits.

The status register should be cleared before attempting the next operation. Any CUI instruction can follow after erasure is completed; however, it must be recognized that reads from the memory array, status register, or Intelligent Identifier can not be accomplished until the CUI is given the appropriate command. A Read Array command must first be given before memory contents can be read.

Figure 8 shows a system software flowchart for Block Erase operation.

3.3.5.1 Suspending and Resuming Erase

Since an erase operation typically requires 1.5 to 3 seconds to complete, an Erase Suspend command is provided. This allows erase-sequence interruption in order to read data from another block of the memory. Once the erase sequence is started, writing the Erase Suspend command to the CUI requests that the Write State Machine (WSM) pause the erase sequence at a predetermined point in the erase algorithm. The status register must be read to determine when the erase operation has been suspended.

At this point, a Read Array command can be written to the CUI in order to read data from blocks other than that which is being suspended. The only other valid command at this time is the Erase Resume command or Read Status Register operation.

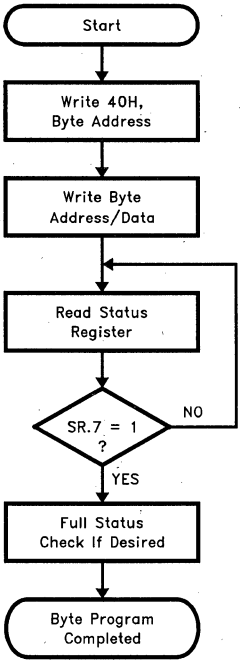
Figure 9 shows a system software flowchart detailing the operation.

During Erase Suspend mode, the chip can go into a pseudo-standby mode by taking CE# to V_{IH} and the active current is now a maximum of 10 mA. If the chip is enabled while in this mode by taking CE# to V_{IL}, the Erase Resume command can be issued to resume the erase operation.

Upon completion of reads from any block other than the block being erased, the Erase Resume command must be issued. When the Erase Resume command is given, the WSM will continue with the erase sequence and complete erasing the block. As with the end of erase, the status register must be read, cleared, and the next instruction issued in order to continue.

3.4.6 EXTENDED CYCLING

Intel has designed extended cycling capability into its ETOX III flash memory technology. The 2-Mbit boot block flash family is designed for 1,000 program/erase cycles on each of the five blocks. The combination of low electric fields, clean oxide processing and minimized oxide area per memory cell subjected to the tunneling electric field, results in very high cycling capability.



290500-5

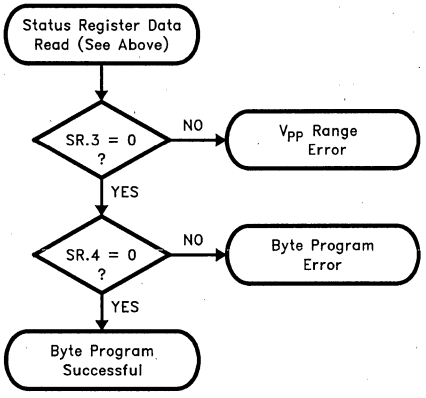
Bus Operation	Command	Comments
Write	Setup Program	Data = 40H Address = Byte to be programmed
Write	Program	Data to be programmed Address = Byte to be programmed
Read		Status Register Data. Toggle OE# or CE# to update Status Register
Standby		Check SR.7 1 = Ready, 0 = Busy

Repeat for subsequent bytes.

Full status check can be done after each byte or after a sequence of bytes.

Write FFH after the last byte programming operation to reset the device to Read Array Mode.

Full Status Check Procedure



290500-6

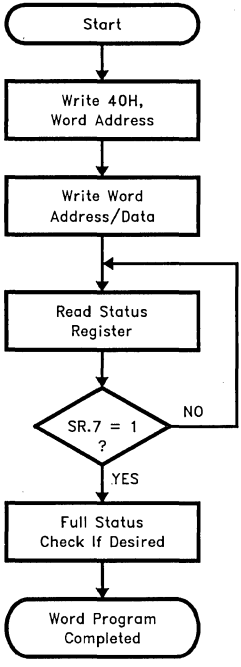
Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4 1 = Byte Program Error

SR.3 MUST be cleared, if set during a program attempt, before further attempts are allowed by the Write State Machine.

SR.4 is only cleared by the Clear Status Register Command, in cases where multiple bytes are programmed before full status is checked.

If error is detected, clear the Status Register before attempting retry or other error recovery.

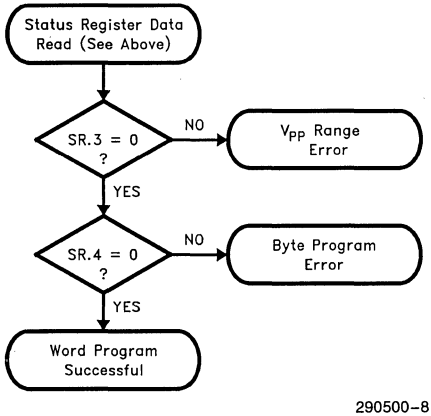
Figure 6. Automated Byte Programming Flowchart



290500-7

Bus Operation	Command	Comments
Write	Setup Program	Data = 40H Address = Word to be programmed
Write	Program	Data to be programmed Address = Word to be programmed
Read		Status Register Data. Toggle OE# or CE# to update Status Register
Standby		Check SR.7 1 = Ready, 0 = Busy
Repeat for subsequent words. Full status check can be done after each word or after a sequence of words. Write FFH after the last word programming operation to reset the device to Read Array Mode.		

Full Status Check Procedure

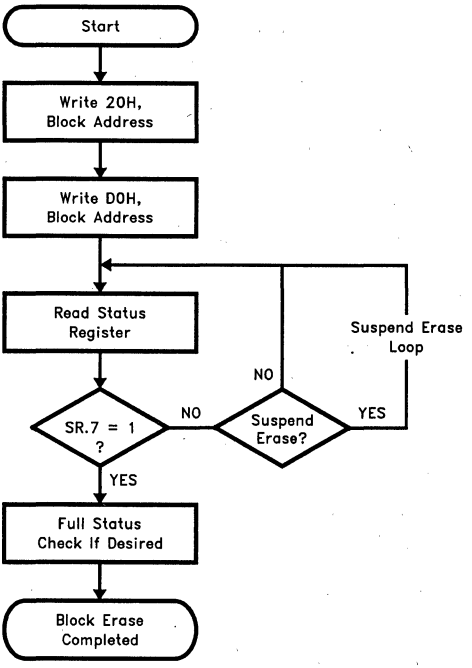


290500-8

Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4 1 = Word Program Error
SR.3 MUST be cleared, if set during a program attempt, before further attempts are allowed by the Write State Machine. SR.4 is only cleared by the Clear Status Register Command, in cases where multiple words are programmed before full status is checked. If error is detected, clear the Status Register before attempting retry or other error recovery.		

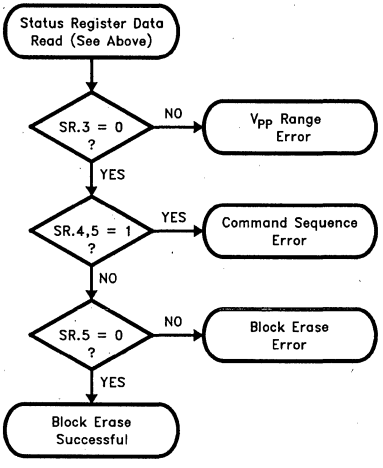
Figure 7. Automated Word Programming Flowchart

8



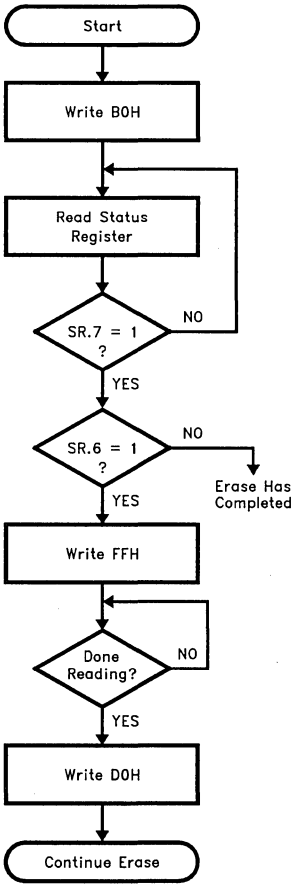
Bus Operation	Command	Comments
Write	Setup Erase	Data = 20H Address = Within block to be erased
Write	Erase	Data = D0H Address = Within block to be erased
Read		Status Register Data. Toggle OE# or CE# to update Status Register
Standby		Check SR.7 1 = Ready, 0 = Busy
Repeat for subsequent blocks. Full status check can be done after each block or after a sequence of blocks. Write FFH after the last block erase operation to reset the device to Read Array Mode.		

Full Status Check Procedure



Bus Operation	Command	Comments
Standby		Check SR.3 1 = V_{PP} Low Detect
Standby		Check SR.4,5 Both 1 = Command Sequence Error
Standby		Check SR.5 1 = Block Erase Error
SR.3 MUST be cleared, if set during an erase attempt, before further attempts are allowed by the Write State Machine. SR.5 is only cleared by the Clear Status Register Command, in cases where multiple blocks are erased before full status is checked. If error is detected, clear the Status Register before attempting retry or other error recovery.		

Figure 8. Automated Block Erase Flowchart



290500-11

Bus Operation	Command	Comments
Write	Erase Suspend	Data = B0H
Read		Status Register Data. Toggle OE# or CE# to update Status Register
Standby		Check SR.7 1 = Ready
Standby		Check SR.6 1 = Suspended
Write	Read Array	Data = FFH
Read		Read array data from block other than that being erased.
Write	Erase Resume	Data = D0H

Figure 9. Erase Suspend/Resume Flowchart

3.4 Power Consumption

3.4.1 ACTIVE POWER

With CE# at a logic-low level and RP# at a logic-high level, the device is placed in the active mode. The device I_{CC} current is a maximum of 65 mA at 10 MHz with TTL input signals.

3.4.2 AUTOMATIC POWER SAVINGS

Automatic Power Savings (APS) is a low power feature during active mode of operation. The 2-Mbit family of products incorporate Power Reduction Control (PRC) circuitry which basically allows the device to put itself into a low current state when it is not being accessed. After data is read from the memory array, PRC logic controls the device's power consumption by entering the APS mode where maximum I_{CC} current is 3 mA and typical I_{CC} current is 1 mA. The device stays in this static state with outputs valid until a new location is read.

3.4.3 STANDBY POWER

With CE# at a logic-high level (V_{IH}), and the CUI in read mode, the memory is placed in standby mode where the maximum I_{CC} standby current is 100 μA with CMOS input signals. The standby operation disables much of the device's circuitry and substantially reduces device power consumption. The outputs (DQ[0:15] or DQ[0:7]) are placed in a high-impedance state independent of the status of the OE# signal. When the 2-Mbit boot block flash family is deselected during erase or program functions, the devices will continue to perform the erase or program function and consume program or erase active power until program or erase is completed.

8

3.4.4 RESET/DEEP POWER-DOWN

The 2-Mbit boot block flash family has a RP# pin which places the device in the deep power-down mode. When RP# is at a logic-low (GND ±0.2V), all circuits are turned off and the device typically draws a maximum 80 μA of V_{CC} current.

During read modes, the RP# pin going low deselects the memory and places the output drivers in a high impedance state. Recovery from the deep power-down state, requires a minimum of 300 ns to access valid data (t_{PHQV}).

During erase or program modes, RP# low will abort either erase or program operation. The contents of the memory are no longer valid as the data has been corrupted by the RP# function. As in the read mode above, all internal circuitry is turned off to achieve the low current level.

RP# transitions to V_{IL} or turning power off to the device will clear the status register.

This use of RP# during system reset is important with automated write/erase devices. When the system comes out of reset it expects to read from the flash memory. Automated flash memories provide status information when accessed during write/erase modes. If a CPU reset occurs with no flash memory reset, proper CPU initialization would not occur because the flash memory would be providing the status information instead of array data. Intel's Flash Memories allow proper CPU initialization following a system reset through the use of the RP# input. In this application RP# is controlled by the same RESET# signal that resets the system CPU.

3.5 Power-Up Operation

The 2-Mbit boot block flash family is designed to offer protection against accidental block erasure or programming during power transitions. Upon power-up the 2-Mbit boot block flash family is indifferent as to which power supply, V_{PP} or V_{CC}, powers-up first. Power suppy sequencing is not required.

The 2-Mbit boot block flash family ensures the CUI is reset to the read mode on power-up.

In addition, on power-up the user must either drop CE# low or present a new address to ensure valid data at the outputs.

A system designer must guard against spurious writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and CE# must be low for a command write, driving either signal to V_{IH} will inhibit writes to the device. The CUI architecture provides an added level of protection since alteration of memory contents can only occur after successful completion of the two-step command sequences. Finally, the device is disabled until RP# is brought to V_{IH}, regardless of the state of its control inputs. This feature provides yet another level of memory protection.

3.6 Power Supply Decoupling

Flash memory's power switching characteristics require careful device decoupling methods. System designers are interested in 3 supply current issues:

- Standby current levels (I_{CCS})
- Active current levels (I_{CCR})
- Transient peaks produced by falling and rising edges of CE#.

Transient current magnitudes depend on the device outputs' capacitive and inductive loading. Two-line control and proper decoupling capacitor selection will suppress these transient voltage peaks. Each flash device should have a 0.1 μF ceramic capacitor connected between each V_{CC} and GND, and between its V_{PP} and GND. These high frequency, low-inherent inductance capacitors should be placed as close as possible to the package leads.

3.6.1 V_{PP} TRACE ON PRINTED CIRCUIT BOARDS

Writing to flash memories while they reside in the target system, requires special consideration of the V_{PP} power supply trace by the printed circuit board designer. The V_{PP} pin supplies the flash memory cell's current for programming and erasing. One should use similar trace widths and layout considerations given to the V_{CC} power supply trace. Adequate V_{PP} supply traces and decoupling will decrease spikes and overshoots.

3.6.2 V_{CC}, V_{PP} AND RP# TRANSITIONS

The CUI latches commands as issued by system software and is not altered by V_{PP} or CE# transitions or WSM actions. Its state upon power-up, after exit from deep power-down mode or after V_{CC} transitions below V_{LKO} (Lockout voltage), is Read Array mode.

After any word/byte write or block erase operation is complete and even after V_{PP} transitions down to V_{PPL}, the CUI must be reset to Read Array mode via the Read Array command when accesses to the flash memory are desired.

ABSOLUTE MAXIMUM RATINGS*

Automatic Operating Temperature
During Read ..... −40°C to +125°C
During Block Erase
and Word/Byte Write ..... −40°C to +125°C
Temperature Under Bias ..... −40°C to +125°C

Storage Temperature ..... −65°C to +150°C

Voltage on Any Pin
(except V_{CC}, A_9, V_{PP} and RP#)
with Respect to GND ..... −2.0V to +7.0V(2)

Voltage on Pin RP# or Pin A_9
with Respect to GND ..... −2.0V to +13.5V(2, 3)

V_{PP} Program Voltage with Respect
to GND during Block Erase
and Word/Byte Write ..... −2.0V to +14.0V(2, 3)

V_{CC} Supply Voltage
with Respect to GND ..... −2.0V to +7.0V(2)

Output Short Circuit Current..... 100 mA(4)

NOTICE: This data sheet contains information on products in the sampling and initial production phases of development. The specifications are subject to change without notice. Verify with your local Intel Sales office that you have the latest data sheet before finalizing a design.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for automotive product defined by this specification.
2. Minimum DC voltage is −0.5V on input/output pins. During transitions, this level may undershoot to −2.0V for periods <20 ns. Maximum DC voltage on input/output pins is V_{CC} + 0.5V which, during transitions, may overshoot to V_{CC} + 2.0V for periods <20 ns.
3. Maximum DC voltage on V_{PP} may overshoot to +14.0V for periods <20 ns. Maximum DC voltage on RP# or A_9 may overshoot to 13.5V for periods <20 ns.
4. Output shorted for no more than one second. No more than one output shorted at a time.

OPERATING CONDITIONS

Symbol	Parameter	Notes	Min	Max	Units
T_A	Operating Temperature		−40	125	°C
V_{CC}	V_{CC} Supply Voltage (10%)	5	4.50	5.50	V

DC CHARACTERISTICS

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Condition
I_{LI}	Input Load Current	1			±1.0	µA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or GND
I_{LO}	Output Leakage Current	1			±10	µA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or GND
I_{CCS}	V_{CC} Standby Current	1, 3			1.5	mA	$V_{CC} = V_{CC}$ Max CE# = RP# = V_{IH}
					130	µA	$V_{CC} = V_{CC}$ Max CE# = RP# = V_{CC} ±0.2V A28F200BX: BYTE# = V_{CC} ±0.2V or GND

8

DC CHARACTERISTICS (Continued)

Symbol	Parameter	Notes	Min	Typ	Max	Unit	Test Condition
I_{CCD}	V_{CC} Deep Power-Down Current	1			80	µA	RP# = GND ±0.2V
I_{CCR}	V_{CC} Read Current for A28F200BX Byte-Wide and Word-Wide Mode	1, 5, 6			60	mA	V_{CC} = V_{CC} Max, CE# = GND f = 10 MHz, I_{OUT} = 0 mA CMOS Inputs
					65	mA	V_{CC} = V_{CC} Max, CE# = V_{IL} f = 10 MHz, I_{OUT} = 0 mA TTL Inputs
I_{CCW}	V_{CC} Write Current	1, 4			65	mA	Word Write in Progress
I_{CCE}	V_{CC} Block Erase Current	1, 4			30	mA	Block Erase in Progress
I_{CCES}	V_{CC} Erase Suspend Current	1, 2		5	10	mA	Block Erase Suspended, CE# = V_{IH}
I_{PPS}	V_{PP} Standby Current	1			±0.15	µA	$V_{PP} \leq V_{CC}$
I_{PPD}	V_{PP} Deep Power-Down Current	1			5.0	µA	RP# = GND ±0.2V
I_{PPR}	V_{PP} Read Current	1			200	µA	$V_{PP} > V_{CC}$
I_{PPW}	V_{PP} Word Write Current	1, 4			40	mA	V_{PP} = V_{PPH} Word Write in Progress
I_{PPW}	V_{PP} Byte Write Current	1, 4			30	mA	V_{PP} = V_{PPH} Byte Write in Progress
I_{PPE}	V_{PP} Block Erase Current	1, 4			30	mA	V_{PP} = V_{PPH} Block Erase in Progress
I_{PPES}	V_{PP} Erase Suspend Current	1			200	µA	V_{PP} = V_{PPH} Block Erase Suspended
$I_{RP\#}$	RP# Current	1, 4			500	µA	RP# = V_{HH}
I_{ID}	A_9 Intelligent Identifier Current	1, 4			500	µA	A_9 = V_{ID}
V_{ID}	A_9 Intelligent Identifier Voltage		11.5		13.0	V	
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.0		V_{CC} + 0.5	V	
V_{OL}	Output Low Voltage				0.45	V	V_{CC} = V_{CC} Min I_{OL} = 5.8 mA
V_{OH}	Output High Voltage		2.4			V	V_{CC} = V_{CC} Min I_{OH} = −2.5 mA
V_{PPL}	V_{PP} during Normal Operations	3	0.0		6.5	V	
V_{PPH}	V_{PP} during Erase/Write Operations	7	11.4	12.0	12.6	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.0			V	
V_{HH}	RP# Unlock Voltage		11.5		13.0	V	Boot Block Write/Erase

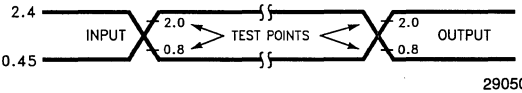
CAPACITANCE[4] T_A = 25°C, f = 1 MHz

Symbol	Parameter	Typ	Max	Unit	Condition
C_{IN}	Input Capacitance	6	8	pF	V_{IN} = 0V
C_{OUT}	Output Capacitance	10	12	pF	V_{OUT} = 0V

NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C. These currents are valid for all product versions (packages and speeds).
2. I_{CCES} is specified with the device deselected. If the device is read while in Erase Suspend Mode, current draw is the sum of I_{CCES} and I_{CCR}.
3. Block Erases and Word/Byte Writes are inhibited when V_{PP} = V_{PPL} and not guaranteed in the range between V_{PPH} and V_{PPL}.
4. Sampled, not 100% tested.
5. Automatic Power Savings (APS) reduces I_{CCR} to less than 1 mA typical in static operation.
6. CMOS Inputs are either V_{CC} ±0.2V or GND ±0.2V. TTL Inputs are either V_{IL} or V_{IH}.
7. V_{PP} = 12.0V ±5% for applications requiring 1,000 block erase cycles.

STANDARD TEST CONFIGURATION

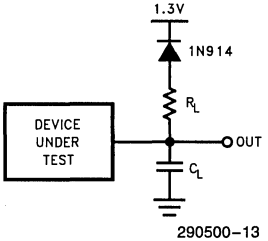
STANDARD AC INPUT/OUTPUT REFERENCE WAVEFORM



290500-12

AC test inputs are driven at V_{OH} (2.4 V_{TTL}) for a Logic "1" and V_{OL} (0.45 V_{TTL}) for a logic "0". Input timing begins at V_{IH} (2.0 V_{TTL}) and V_{IL} (0.8 V_{TTL}). Output timing ends at V_{IH} and V_{IL}. Input rise and fall times (10% to 90%) < 10 ns.

STANDARD AC TESTING LOAD CIRCUIT



290500-13

C_L = 100 pF
C_L Includes Jig Capacitance
R_L = 3.3 KΩ

8

AC CHARACTERISTICS—Read Only Operations[1]:

Versions				A28F200BX-90[4]		Unit
Symbol		Parameter	Notes	Min	Max	
t_{AVAV}	t_{RC}	Read Cycle Time		90		ns
t_{AVQV}	t_{ACC}	Address to Output Delay			90	ns
t_{ELQV}	t_{CE}	CE# to Output Delay	2		90	ns
t_{PHQV}	t_{PWH}	RP# High to Output Delay			300	ns
t_{GLQV}	t_{OE}	OE# to Output Delay	2		45	ns
t_{ELQX}	t_{LZ}	CE# to Output Low Z	3	0		ns
t_{EHQZ}	t_{HZ}	CE# High to Output High Z	3		35	ns
t_{GLQX}	t_{OLZ}	OE# to Output Low Z	3	0		ns
t_{GHQZ}	t_{DF}	OE# High to Output High Z	3		35	ns
	t_{OH}	Output Hold from Addresses, CE# or OE# Change, Whichever is First	3	0		ns
t_{ELFL} t_{ELFH}		CE# to BYTE# Switching Low or High	3		5	ns
t_{FHQV}		BYTE# Switching High to Valid Output Delay	3, 5		90	ns
t_{FLQZ}		BYTE# Switching Low to Output High Z	3		35	ns

NOTES:
1. See AC Input/Output Reference Waveform for timing measurements.
2. OE# may be delayed up to $t_{CE}-t_{OE}$ after the falling edge of CE# without impact on t_{CE}.
3. Sampled, not 100% tested.
4. See Standard Test Configuration.
5. t_{FLQV}, BYTE# switching low to valid output delay, will be equal to t_{AVQV}, measured from the time $DQ_5/A\text{-}_1$ becomes valid.

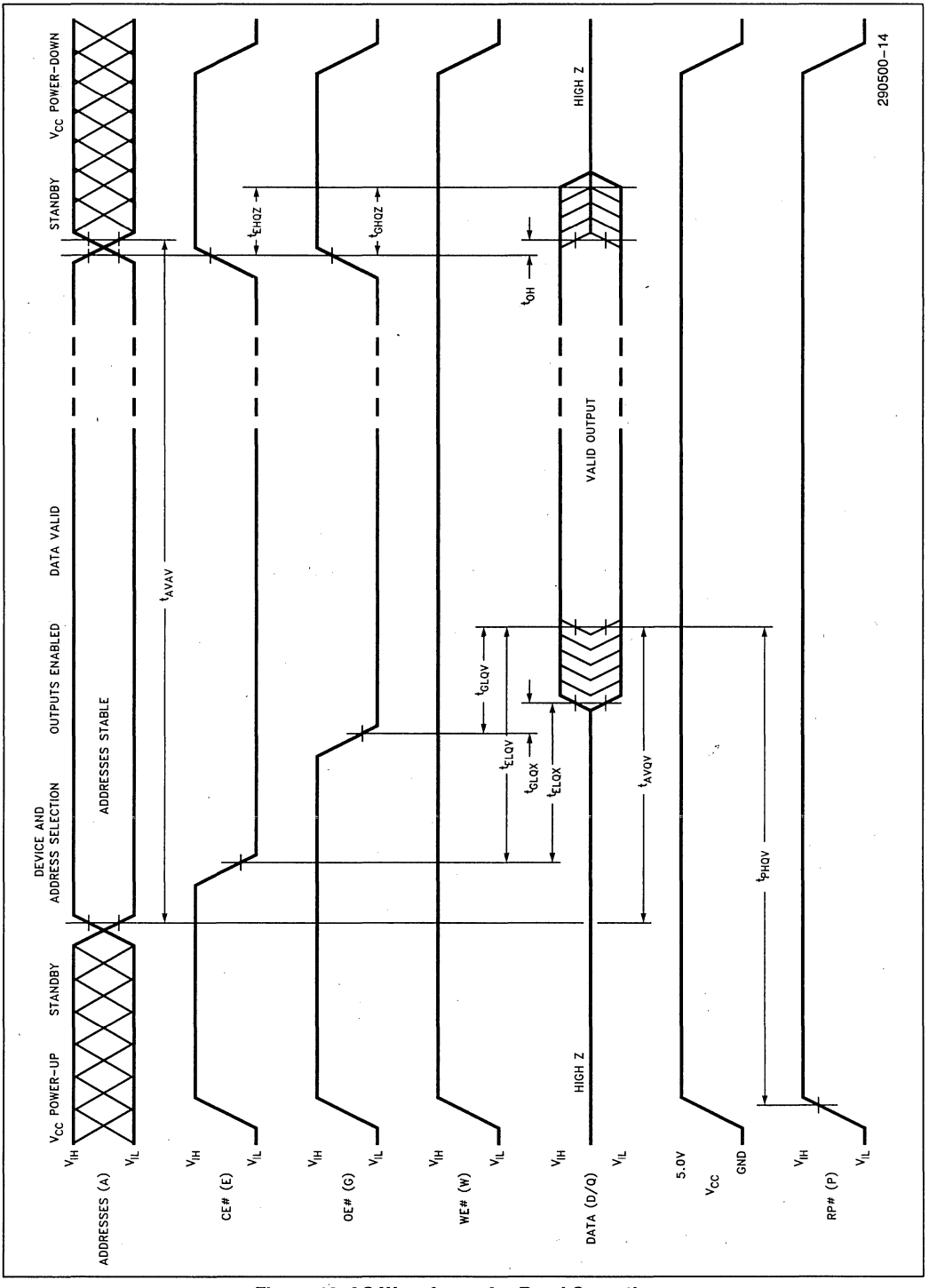


Figure 10. AC Waveforms for Read Operations

8

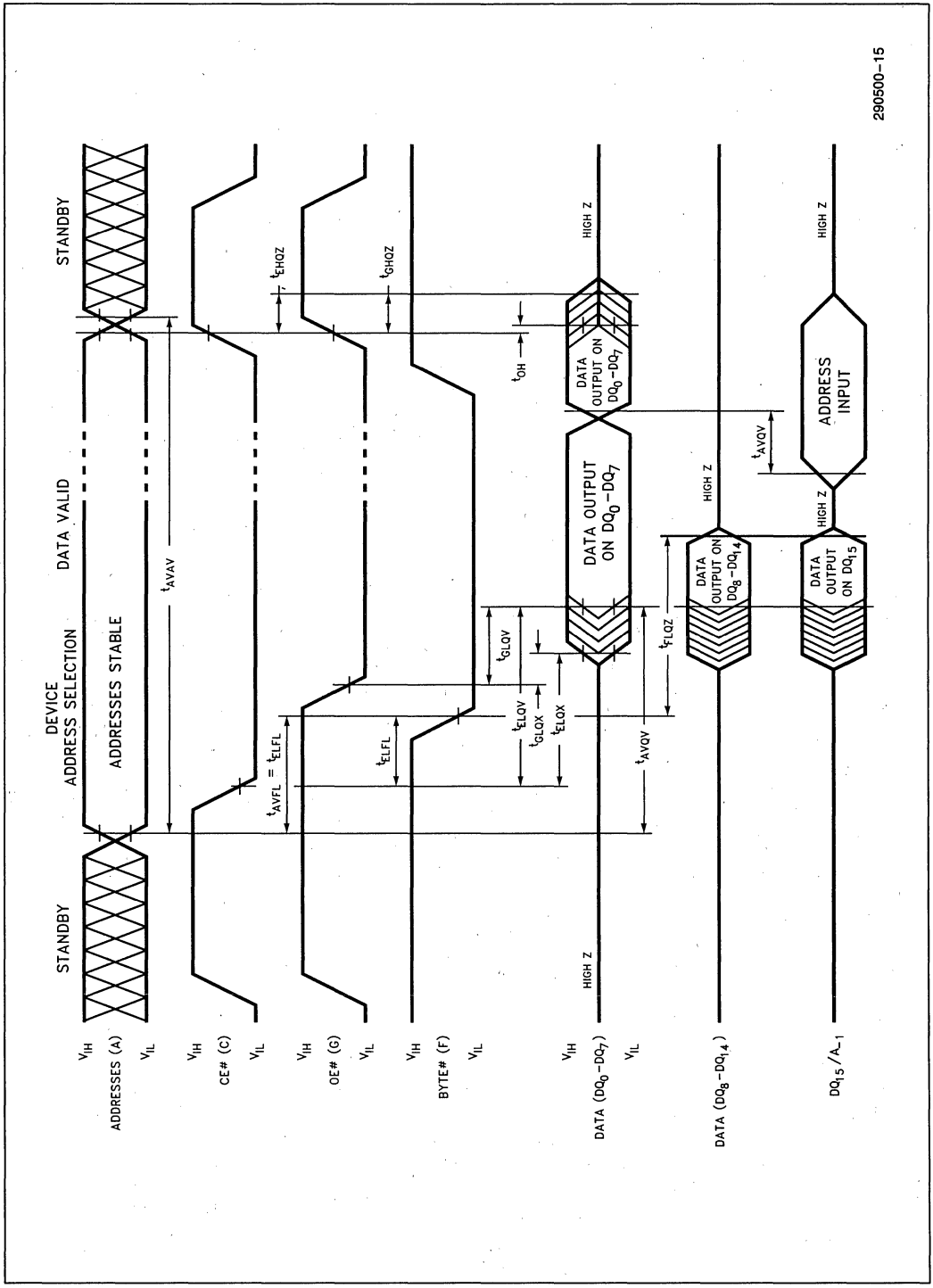


Figure 11. BYTE# Timing for Both Read and Write Operations for A28F200BX

AC CHARACTERISTICS For WE#-Controlled Write Operations[1]:

Versions[4]				A28F200BX-90[9]		Unit
Symbol		Parameter	Notes	Min	Max	
t_{AVAV}	t_{WC}	Write Cycle Time		90		ns
t_{PHWL}	t_{PS}	RP# High Recovery to WE# Going Low		210		ns
t_{ELWL}	t_{CS}	CE# Setup to WE# Going Low		0		ns
t_{PHHWH}	t_{PHS}	RP# V_{HH} Setup to WE# Going High	6, 8	100		ns
t_{VPWH}	t_{VPS}	V_{PP} Setup to WE# Going High	5, 8	100		ns
t_{AVWH}	t_{AS}	Address Setup to WE# Going High	3	60		ns
t_{DVWH}	t_{DS}	Data Setup to WE# Going High	4	60		ns
t_{WLWH}	t_{WP}	WE# Pulse Width		60		ns
t_{WHDX}	t_{DH}	Data Hold from WE# High	4	0		ns
t_{WHAX}	t_{AH}	Address Hold from WE# High	3	10		ns
t_{WHEH}	t_{CH}	CE# Hold from WE# High		10		ns
t_{WHWL}	t_{WPH}	WE# Pulse Width High		30		ns
t_{WHQV1}		Duration of Word/Byte Write Operation	2, 5	7		µs
t_{WHQV2}		Duration of Erase Operation (Boot)	2, 5, 6	0.4		s
t_{WHQV3}		Duration of Erase Operation (Parameter)	2, 5	0.4		s
t_{WHQV4}		Duration of Erase Operation (Main)	2, 5, 6	0.7		s
t_{QWL}	t_{VPH}	V_{PP} Hold from Valid SRD	5, 8	0		ns
t_{QVPH}	t_{PHH}	RP# V_{HH} Hold from Valid SRD	6, 8	0		ns
t_{PHBR}		Boot-Block Relock Delay	7, 8		100	ns

NOTES:
1. Read timing characteristics during write and erase operations are the same as during read-only operations. Refer to AC Characteristics during Read Mode.
2. The on-chip WSM completely automates program/erase operations; program/erase algorithms are now controlled internally which includes verify and margining operations.
3. Refer to command definition table for valid A_{IN}.
4. Refer to command definition table for valid D_{IN}.
5. Program/Erase durations are measured to valid SRD data (successful operation, SR.7 = 1).
6. For Boot Block Program/Erase, RP# should be held at V_{HH} until operation completes successfully.
7. Time t_{PHBR} is required for successful relocking of the Boot Block.
8. Sampled but not 100% tested.
9. See Standard Test Configuration.

BLOCK ERASE AND WORD/BYTE WRITE PERFORMANCE: V_{PP} = 12.0V ±5%

Parameter	Notes	A28F200BX-90			Unit
		Min	Typ(1)	Max	
Boot/Parameter Block Erase Time	2		1.5	10.5	s
Main Block Erase Time	2		3.0	18	s
Main Block Byte Program Time	2		1.4	5.0	s
Main Block Word Program Time	2		0.7	2.5	s

NOTES:
1. 25°C, 12.0V V_{PP}.
2. Excludes System-Level Overhead.

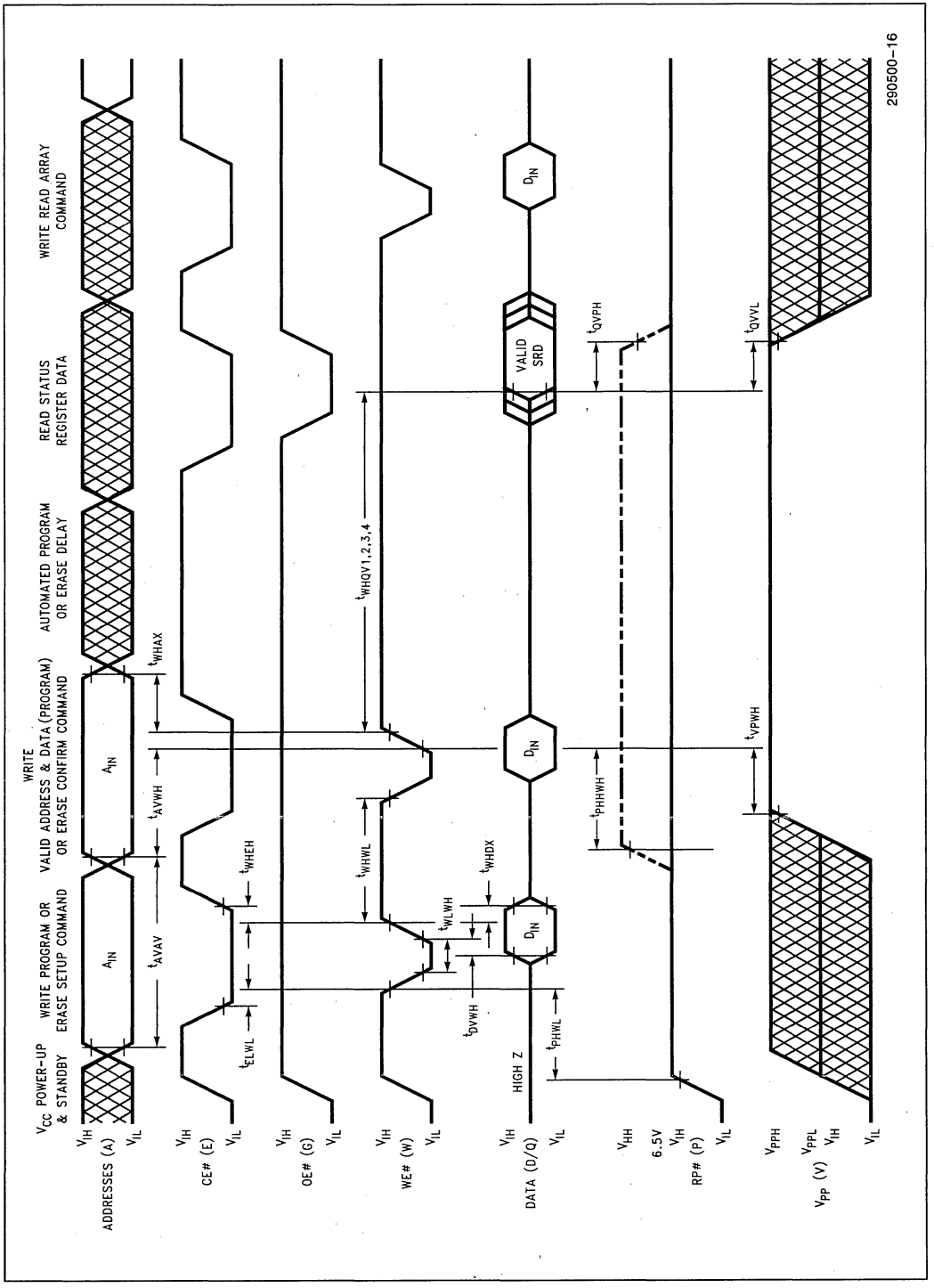


Figure 12. AC Waveforms for Write and Erase Operations (WE#-Controlled Writes)

8

AC CHARACTERISTICS FOR CE#-CONTROLLED WRITE OPERATIONS(1, 9)

Versions				A28F200BX-90(10)		Unit
Symbol		Parameter	Notes	Min	Max	
t_{AVAV}	t_{WC}	Write Cycle Time		90		ns
t_{PHEL}	t_{PS}	RP# High Recovery to CE# Going Low		210		ns
t_{WLEL}	t_{WS}	WE# Setup to CE# Going Low		0		ns
t_{PHHEH}	t_{PHS}	RP# V_{HH} Setup to CE# Going High	6, 8	100		ns
t_{VPEH}	t_{VPS}	V_{PP} Setup to CE# Going High	5, 8	100		ns
t_{AVEH}	t_{AS}	Address Setup to CE# Going High	3	60		ns
t_{DVEH}	t_{DS}	Data Setup to CE# Going High	4	60		ns
t_{ELEH}	t_{CP}	CE# Pulse Width		60		ns
t_{EHDX}	t_{DH}	Data Hold from CE# High	4	0		ns
t_{EHAX}	t_{AH}	Address Hold from CE# High	3	10		ns
t_{EHWH}	t_{WH}	WE# Hold from CE# High		10		ns
t_{EHEL}	t_{CPH}	CE# Pulse Width High		30		ns
t_{EHQV1}		Duration of Word/Byte Programming Operation	2, 5	7		μs
t_{EHQV2}		Duration of Erase Operation (Boot)	2, 5, 6	0.4		s
t_{EHQV3}		Duration of Erase Operation (Parameter)	2, 5	0.4		s
t_{EHQV4}		Duration of Erase Operation (Main)	2, 5	0.7		s
t_{QWL}	t_{VPH}	V_{PP} Hold from Valid SRD	5, 8	0		ns
t_{QVPH}	t_{PHH}	RP# V_{HH} Hold from Valid SRD	6, 8	0		ns
t_{PHBR}		Boot-Block Relock Delay	7		100	ns

NOTES:
1. Chip-Enable Controlled Writes: Write operations are driven by the valid combination of CE# and WE# in systems where CE# defines the write pulse-width (within a longer WE# timing waveform), all set-up, hold and inactive WE# time should be measured relative to the CE# waveform.
2, 3, 4, 5, 6, 7, 8: Refer to AC Characteristics for WE#-Controlled Write Operations.
9. Read timing characteristics during write and erase operations are the same as during read-only operations. Refer to AC Characteristics during Read Mode.
10. See Standard Test Configuration.

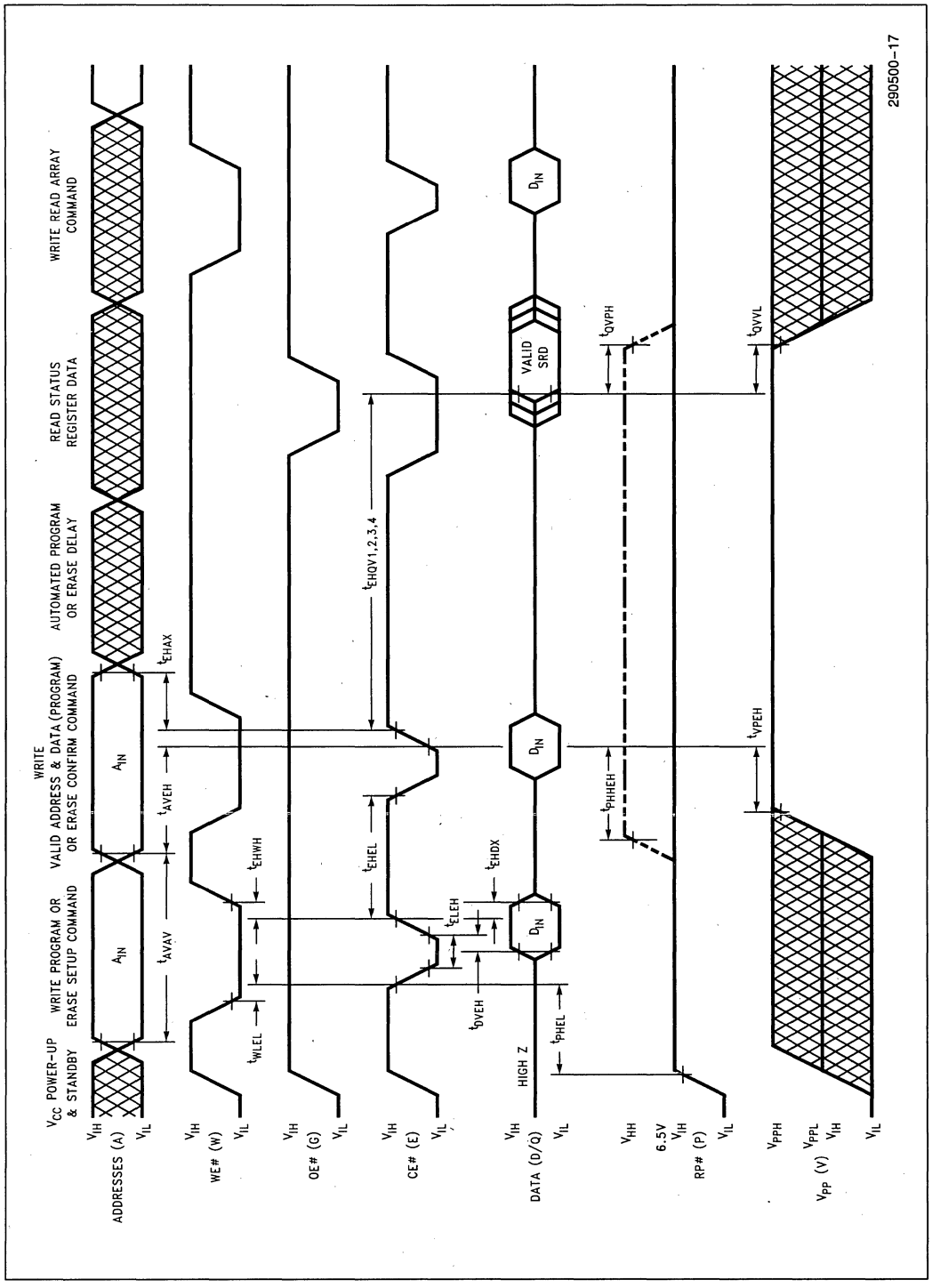
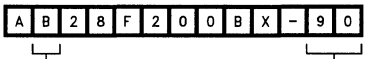


Figure 13. Alternate AC Waveforms for Write and Erase Operations (CE#-Controlled Writes)

ORDERING INFORMATION



290500-18

Valid Combinations:
AB28F200BX-T90 AB28F200BX-B90

ADDITIONAL INFORMATION

	Order Number
A28F400BX Datasheet	290501
28F400BX/28F004BX Datasheet	290451
A28F200BX-L/28F002BX-L Datasheet	290449
28F400BX-L/28F004BX-L Datasheet	290450
AP-363 "Extended Flash BIOS Design for Portable Computers"	292098
ER-28 "ETOX™ III Flash Memory Technology"	204012
ER-29 "The Intel 2/4-Mbit Boot Block Flash Memory Family"	294013

REVISION HISTORY

Number	Description
002	Changed Package Designator

A28F010
1024K (128K x 8) CMOS FLASH MEMORY
(Automotive)

- **Automotive Temperature Range: −40°C to +125°C**
- **Flash Memory Electrical Chip-Erase — 1 Second Typical Chip-Erase**
- **Quick-Pulse Programming Algorithm — 10 μs Typical Byte-Program — 2 Second Chip-Program**
- **1,000 Erase/Program Cycles Minimum over Automotive Temperature Range**
- **12.0V ±5% V_{PP}**
- **High-Performance Read — 120 ns Maximum Access Time**
- **CMOS Low Power Consumption — 30 mA Maximum Active Current — 300 μA Maximum Standby Current**
- **Integrated Program/Erase Stop Timer**
- **Command Register Architecture for Microprocessor/Microcontroller Compatible Write Interface**
- **Noise Immunity Features — ±10% V_{CC} Tolerance — Maximum Latch-Up Immunity through EPI Processing**
- **ETOX™ III Flash Nonvolatile Memory Technology — EPROM-Compatible Process Base — High-Volume Manufacturing Experience**
- **JEDEC-Standard Pinouts — 32-Pin Plastic DIP — 32-Lead PLCC**

(See Packaging Spec., Order #231369)

Intel's 28F010 CMOS flash memory offers the most cost-effective and reliable alternative for read/write random access nonvolatile memory. The 28F010 adds electrical chip-erasure and reprogramming to familiar EPROM technology. Memory contents can be rewritten: in a test socket; in a PROM-programmer socket; on-board during subassembly test; in-system during final test; and in-system after-sale. The 28F010 increases memory flexibility, while contributing to time- and cost-savings.

The 28F010 is a 1024-kilobit nonvolatile memory organized as 131,072 bytes of 8 bits. Intel's 28F010 is offered in 32-pin Plastic DIP or 32-lead PLCC packages. Pin assignments conform to JEDEC standards.

Extended erase and program cycling capability is designed into Intel's ETOX™ III (EPROM Tunnel Oxide) process technology. Advanced oxide processing, an optimized tunneling structure, and lower electric field combine to extend reliable cycling beyond that of traditional EEPROMs. With the 12.0V V_{PP} supply, the 28F010 performs a minimum of 1,000 erase and program cycles well within the time limits of the Quick-Pulse Programming and Quick-Erase algorithms.

Intel's 28F010 employs advanced CMOS circuitry for systems requiring high-performance access speeds, low power consumption, and immunity to noise. Its 120 nanosecond access time provides no-WAIT-state performance for a wide range of microprocessors and microcontrollers. Maximum standby current of 300 μA translates into power savings when the device is deselected. Finally, the highest degree of latch-up protection is achieved through Intel's unique EPI processing. Prevention of latch-up is provided for stresses up to 100 mA on address and data pins, from −1V to V_{CC} + 1V.

With Intel's ETOX III process base, the 28F010 leverages years of EPROM experience to yield the highest levels of quality, reliability, and cost-effectiveness.

In order to meet the rigorous environmental requirements of automotive applications, Intel offers the 28F010 in extended automotive temperature range. Read and write characteristics are guaranteed over the range of −40°C to +125°C ambient.

November 1994
Order Number: 290266-004

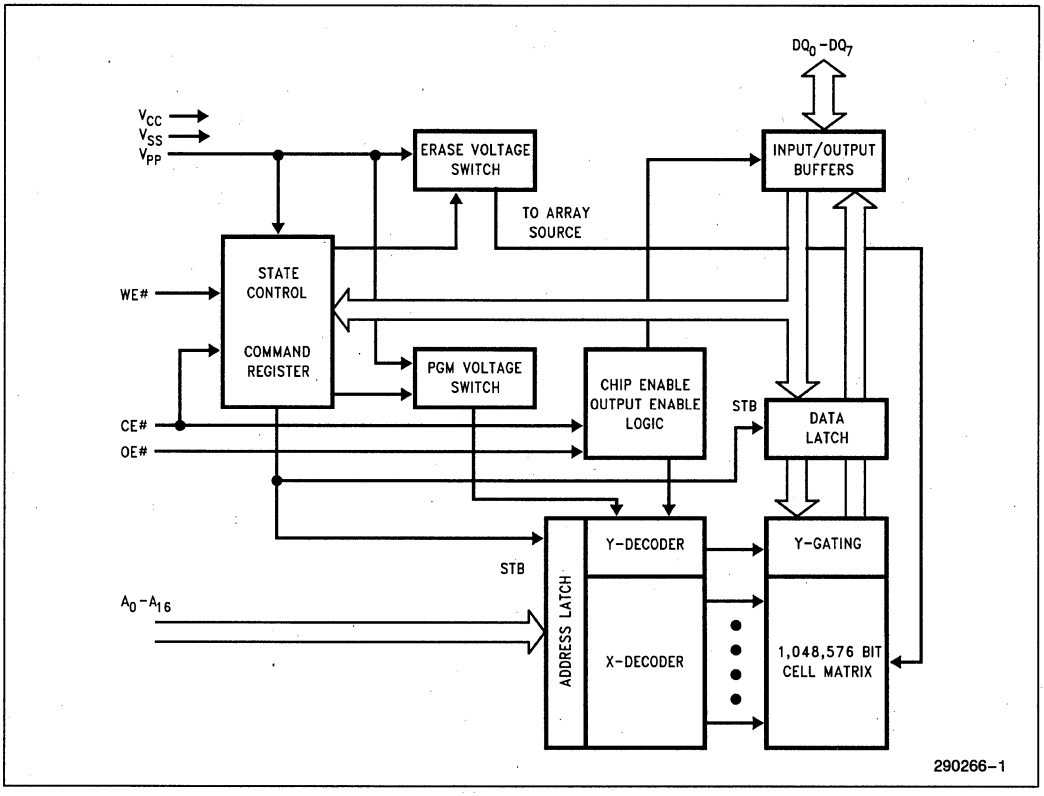


Figure 1. 28F010 Block Diagram

AUTOMOTIVE TEMPERATURE FLASH MEMORIES

The Intel Automotive Flash memories have received additional processing to enhance product characteristics. The automotive temperature range is −40°C to +125°C during the read/write/erase/program operations.

Speed Versions	Packaging Options	
	Plastic DIP	PLCC
150	AP	AN
120	AP	AN

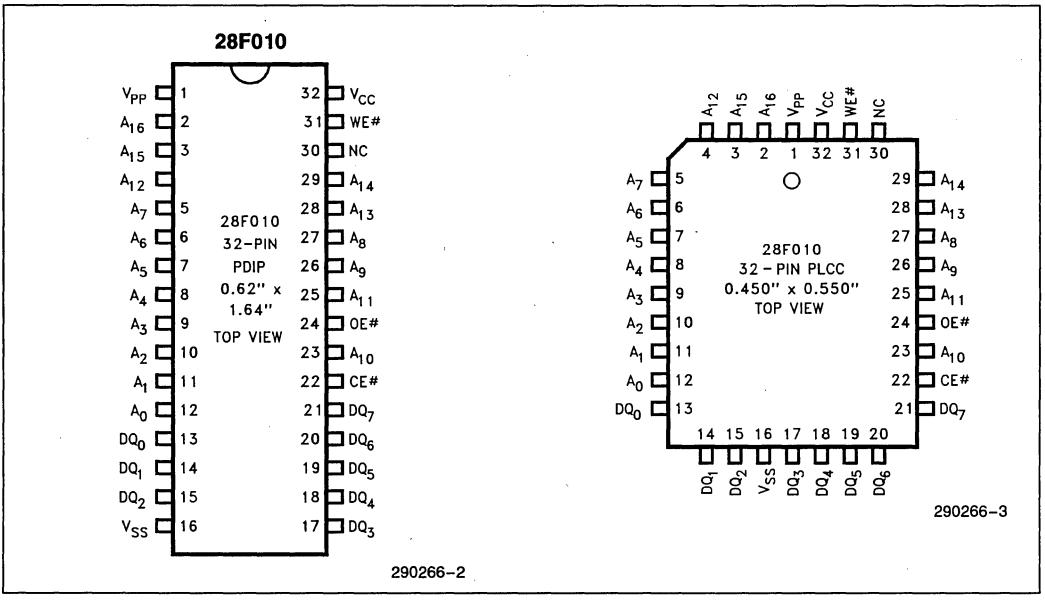


Figure 2. 28F010 Pin Configurations

Table 1. Pin Description

Symbol	Type	Name and Function
A_0–A_{16}	INPUT	**ADDRESS INPUTS** for memory addresses. Addresses are internally latched during a write cycle.
DQ_0–DQ_7	INPUT/OUTPUT	**DATA INPUT/OUTPUT:** Inputs data during memory write cycles; outputs data during memory read cycles. The data pins are active high and float to tri-state OFF when the chip is deselected or the outputs are disabled. Data is internally latched during a write cycle.
CE#	INPUT	**CHIP ENABLE:** Activates the device's control logic, input buffers, decoders and sense amplifiers. CE# is active low; CE# high deselects the memory device and reduces power consumption to standby levels.
OE#	INPUT	**OUTPUT ENABLE:** Gates the devices output through the data buffers during a read cycle. OE# is active low.
WE#	INPUT	**WRITE ENABLE:** Controls writes to the control register and the array. Write enable is active low. Addresses are latched on the falling edge and data is latched on the rising edge of the WE# pulse. **Note:** With $V_{PP} \leq 6.5V$, memory contents cannot be altered.
V_{PP}		**ERASE/PROGRAM POWER SUPPLY** for writing the command register, erasing the entire array, or programming bytes in the array.
V_{CC}		**DEVICE POWER SUPPLY** (5V ±10%)
V_{SS}		**GROUND**
NC		**NO INTERNAL CONNECTION** to device. Pin may be driven or left floating.

8

APPLICATIONS

The 28F010 flash-memory adds electrical chip-erasure and reprogrammability to EPROM non-volatility and ease of use. The 28F010 is ideal for storing code or data-tables in applications where periodic updates are required. The 28F010 also serves as a dense, nonvolatile data acquisition and storage medium.

The need for code updates pervades all phases of a system's life—from prototyping to system manufacture to after-sale service. In the factory, during prototyping, revisions to control code necessitate ultraviolet erasure and reprogramming of EPROM-based prototype codes. The 28F010 replaces the 15- to 20-minute ultraviolet erasure with one-second electrical erasure. Electrical chip-erasure and reprogramming occur in the same workstation or PROM-programmer socket.

Diagnostics, performed at subassembly or final assembly stages, often require the socketing of EPROMs. Socketed test codes are ultimately replaced with EPROMs containing the final program. With electrical chip-erasure and reprogramming, the 28F010 is soldered to the circuit board. Test codes are programmed into the 28F010 as it resides on the circuit board. Ultimately, the final code can be downloaded to the device. The 28F010's in-circuit alterability eliminates unnecessary handling and less-reliable socketed connections, while adding greater test flexibility.

Material and labor costs associated with code changes increase at higher levels of system integration—the most costly being code updates after sale. Code "bugs", or the desire to augment system functionality, prompt after-sale code updates. Field revisions to EPROM-based code require the removal of EPROM components or entire boards.

Designing with the in-circuit alterable 28F010 eliminates socketed memories, reduces overall material costs, and drastically cuts the labor costs associated with code updates. With the 28F010, code updates are implemented locally via an edge-connector, or remotely over a serial communication link.

The 28F010's electrical chip-erasure, byte reprogrammability, and complete nonvolatility fit well with data accumulation needs. Electrical chip-erasure gives the designer a "blank-slate" in which to log data. Data can be periodically off-loaded for analysis—erasing the slate and repeating the cycle. Or, multiple devices can maintain a "rolling window" of accumulated data.

With high density, nonvolatility, and extended cycling capability, the 28F010 offers an innovative alternative for mass storage. Integrating main memory and backup storage functions into directly executable flash memory boosts system performance, shrinks system size, and cuts power consumption. Reliability exceeds that of electromechanical media, with greater durability in extreme environmental conditions.

A high degree of on-chip feature integration simplifies memory-to-processor interfacing. Figure 3 depicts two 28F010s tied to the 80C186 system bus. The 28F010's architecture minimizes interface circuitry needed for complete in-circuit updates of memory contents.

With cost-effective in-system reprogramming and extended cycling capability, the 28F010 fills the functionality gap between traditional EPROMs and EEPROMs. EPROM-compatible specifications, straightforward interfacing, and in-circuit alterability allows designers to easily augment memory flexibility and satisfy the need for updatable nonvolatile storage in today's designs.

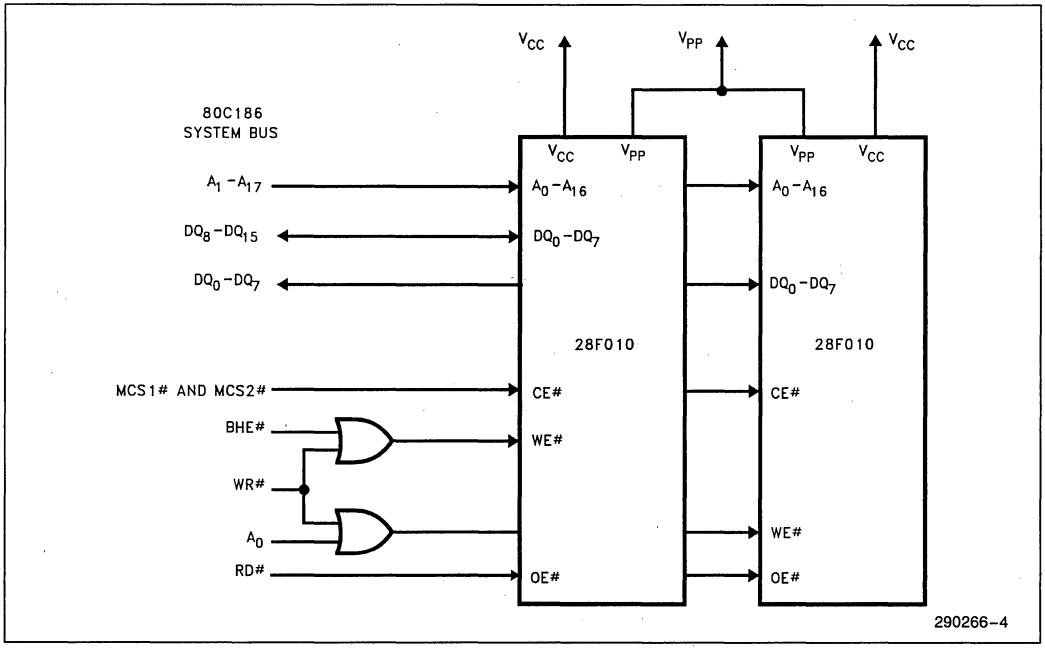


Figure 3. 28F010 in a 80C186 System

PRINCIPLES OF OPERATION

Flash-memory augments EPROM functionality with in-circuit electrical erasure and reprogramming. The 28F010 introduces a command register to manage this new functionality. The command register allows for: 100% TTL-level control inputs; fixed power supplies during erasure and programming; and maximum EPROM compatibility.

In the absence of high voltage on the V_{PP} pin, the 28F010 is a read-only memory. Manipulation of the external memory-control pins yields the standard EPROM read, standby, output disable, and Intelligent Identifier operations.

The same EPROM read, standby, and output disable operations are available when high voltage is applied to the V_{PP} pin. In addition, high voltage on V_{PP} enables erasure and programming of the device. All functions associated with altering memory contents—Intelligent Identifier, erase, erase verify, program, and program verify—are accessed via the command register.

Commands are written to the register using standard microprocessor write timings. Register contents serve as input to an internal state-machine which controls the erase and programming circuitry. Write cycles also internally latch addresses and data needed for programming or erase operations. With the appropriate command written to the register, standard microprocessor read timings output array data, access the Intelligent Identifier codes, or output data for erase and program verification.

Integrated Program/Erase Stop Timer

Successive command write cycles define the durations of program and erase operations; specifically, the program or erase time durations are normally terminated by associated program or erase verify commands. An integrated stop timer provides simplified timing control over these operations; thus eliminating the need for maximum program/erase timing specifications. Program and erase pulse durations are minimums only. When the stop timer terminates a program or erase operation, the device enters an inactive state and remains inactive until receiving the appropriate verify or reset command.

Write Protection

The command register is only alterable when V_{PP} is at high voltage. Depending upon the application, the system designer may choose to make the V_{PP} power supply switchable—available only when memory updates are desired. When high voltage is removed,

8

Table 2. 28F010 Bus Operations

Operation	Pins	V_{PP}(1)	A_0	A_9	CE#	OE#	WE#	DQ_0–DQ_7
READ-ONLY	Read	V_{PPL}	A_0	A_9	V_{IL}	V_{IL}	V_{IH}	Data Out
	Output Disable	V_{PPL}	X	X	V_{IL}	V_{IH}	V_{IH}	Tri-State
	Standby	V_{PPL}	X	X	V_{IH}	X	X	Tri-State
	Intelligent Identifier (Mfr)(2)	V_{PPL}	V_{IL}	V_{ID}(3)	V_{IL}	V_{IL}	V_{IH}	Data = 89H
	Intelligent Identifier (Device)(2)	V_{PPL}	V_{IH}	V_{ID}(3)	V_{IL}	V_{IL}	V_{IH}	Data = B4H
READ/WRITE	Read	V_{PPH}	A_0	A_9	V_{IL}	V_{IL}	V_{IH}	Data Out(4)
	Output Disable	V_{PPH}	X	X	V_{IL}	V_{IH}	V_{IH}	Tri-State
	Standby(5)	V_{PPH}	X	X	V_{IH}	X	X	Tri-State
	Write	V_{PPH}	A_0	A_9	V_{IL}	V_{IH}	V_{IL}	Data In(6)

NOTES:
1. V_{PPL} may be ground, a no-connect with a resistor tied to ground, or ≤ 6.5V. V_{PPH} is the programming voltage specified for the device. Refer to D.C. Characteristics. When $V_{PP} = V_{PPL}$ memory contents can be read but not written or erased.
2. Manufacturer and device codes may also be accessed via a command register write sequence. Refer to Table 3. All other addresses low.
3. V_{ID} is the Intelligent Identifier high voltage. Refer to DC Characteristics.
4. Read operations with $V_{PP} = V_{PPH}$ may access array data or the Intelligent Identifier codes.
5. With V_{PP} at high voltage, the standby current equals $I_{CC} + I_{PP}$ (standby).
6. Refer to Table 3 for valid Data-In during a write operation.
7. X can be V_{IL} or V_{IH}.

the contents of the register default to the read command, making the 28F010 a read-only memory. Memory contents cannot be altered.

Or, the system designer may choose to "hardwire" V_{PP}, making the high voltage supply constantly available. In this instance, all operations are performed in conjunction with the command register. The 28F010 is designed to accommodate either design practice, and to encourage optimization of the processor-memory interface.

The two-step Program/Erase write sequence to the Command Register provides additional software write protection.

BUS OPERATIONS

Read

The 28F010 has two control functions, both of which must be logically active, to obtain data at the outputs. Chip-Enable (CE#) is the power control and should be used for device selection. Output-Enable (OE#) is the output control and should be used to gate data from the output pins, independent of device selection. Figure 6 illustrates read timing waveforms.

When V_{PP} is low (V_{PPL}), the read only operation is active. This permits reading the data in the array and outputting the Intelligent Identifier codes (see Table 2). When V_{PP} is high (V_{PPH}), the default condition of the device is the read only mode. This allows reading the data in the array. Further functionality is achieved though the Command Register as shown in Table 3.

Output Disable

With Output-Enable at a logic-high level (V_{IH}), output from the device is disabled. Output pins are placed in a high-impedance state.

Standby

With Chip-Enable at a logic-high level, the standby operation disables most of the 28F010's circuitry and substantially reduces device power consumption. The outputs are placed in a high-impedance state, independent of the Output-Enable signal. If the 28F010 is deselected during erasure, programming, or program/erase verification, the device draws active current until the operation is terminated.

Intelligent Identifier Operation

The Intelligent Identifier operation outputs the manufacturer code (89H) and device code (B4H). Programming equipment automatically matches the device with its proper erase and programming algorithms.

With Chip-Enable and Output-Enable at a logic low level, raising A9 to high voltage V_{ID} (see DC Characteristics) activates the operation. Data read from locations 0000H and 0001H represent the manufacturer's code and the device code, respectively.

The manufacturer- and device-codes can also be read via the command register, for instances where the 28F010 is erased and reprogrammed in the target system. Following a write of 90H to the command register, a read from address location 0000H outputs the manufacturer code (89H). A read from address 0001H outputs the device code (B4H).

Write

Device erasure and programming are accomplished via the command register, when high voltage is applied to the V_{PP} pin. The contents of the register serve as input to the internal state-machine. The state-machine outputs dictate the function of the device.

The command register itself does not occupy an addressable memory location. The register is a latch used to store the command, along with address and data information needed to execute the command.

The command register is written by bringing Write-Enable to a logic-low level (V_{IL}), while Chip-Enable is low. Addresses are latched on the falling edge of Write-Enable, while data is latched on the rising edge of the Write-Enable pulse. Standard microprocessor write timings are used.

Refer to A.C. Write Characteristics and the Erase/Programming Waveforms for specific timing parameters.

COMMAND DEFINITIONS

When low voltage is applied to the V_{PP} pin, the contents of the command register default to 00H, enabling read-only operations.

Placing high voltage on the V_{PP} pin enables read/write operations. Device operations are selected by writing specific data patterns into the command register. Table 3 defines these 28F010 register commands.

Table 3. Command Definitions

Command	Bus Cycles	First Bus Cycle			Second Bus Cycle		
	Req'd	Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory	1	Write	X	00H			
Read Intelligent Identifier Codes(4)	2	Write	X	90H	Read	IA	ID
Set-up Erase/Erase(5)	2	Write	X	20H	Write	X	20H
Erase Verify(5)	2	Write	EA	A0H	Read	X	EVD
Set-up Program/Program(6)	2	Write	X	40H	Write	PA	PD
Program Verify(6)	2	Write	X	C0H	Read	X	PVD
Reset(7)	2	Write	X	FFH	Write	X	FFH

NOTES:
1. Bus operations are defined in Table 2.
2. IA = Identifier address: 00H for manufacturer code, 01H for device code.
   EA = Address of memory location to be read during erase verify.
   PA = Address of memory location to be programmed.
   Addresses are latched on the falling edge of the Write-Enable pulse.
3. ID = Data read from location IA during device identification (Mfr = 89H, Device = B4H).
   EVD = Data read from location EA during erase verify.
   PD = Data to be programmed at location PA. Data is latched on the rising edge of Write-Enable.
   PVD = Data read from location PA during program verify. PA is latched on the Program command.
4. Following the Read Intelligent ID command, two read operations access manufacturer and device codes.
5. Figure 5 illustrates the Quick-Erase Algorithm.
6. Figure 4 illustrates the Quick-Pulse Programming Algorithm.
7. The second bus cycle must be followed by the desired command register write.

Read Command

While V_{PP} is high, for erasure and programming, memory contents can be accessed via the read command. The read operation is initiated by writing 00H into the command register. Microprocessor read cycles retrieve array data. The device remains enabled for reads until the command register contents are altered.

The default contents of the register upon V_{PP} power-up is 00H. This default value ensures that no spurious alteration of memory contents occurs during the V_{PP} power transition. Where the V_{PP} supply is hard-wired to the 28F010, the device powers-up and remains enabled for reads until the command-register contents are changed. Refer to the A.C. Read Characteristics and Waveforms for specific timing parameters.

Intelligent Identifier Command

Flash-memories are intended for use in applications where the local CPU alters memory contents. As such, manufacturer- and device-codes must be accessible while the device resides in the target system. PROM programmers typically access signature codes by raising A9 to a high voltage. However, multiplexing high voltage onto address lines is not a desired system-design practice.

The 28F010 contains an inteligent Identifier operation to supplement traditional PROM-programming methodology. The operation is initiated by writing 90H into the command register. Following the command write, a read cycle from address 0000H retrieves the manufacturer code of 89H. A read cycle from address 0001H returns the device code of B4H. To terminate the operation, it is necessary to write another valid command into the register.

Set-up Erase/Erase Commands

Set-up Erase is a command-only operation that stages the device for electrical erasure of all bytes in the array. The set-up erase operation is performed by writing 20H to the command register.

To commence chip-erasure, the erase command (20H) must again be written to the register. The erase operation begins with the rising edge of the Write-Enable pulse and terminates with the rising edge of the next Write-Enable pulse (i.e., Erase-Verify Command).

This two-step sequence of set-up followed by execution ensures that memory contents are not accidentally erased. Also, chip-erasure can only occur when high voltage is applied to the V_{PP} pin. In the absence of this high voltage, memory contents are protected against erasure. Refer to A.C. Erase Characteristics and Waveforms for specific timing parameters.

Erase-Verify Command

The erase command erases all bytes of the array in parallel. After each erase operation, all bytes must be verified. The erase verify operation is initiated by writing A0H into the command register. The address for the byte to be verified must be supplied as it is latched on the falling edge of the Write-Enable pulse. The register write terminates the erase operation with the rising edge of its Write-Enable pulse.

The 28F010 applies an internally-generated margin voltage to the addressed byte. Reading FFH from the addressed byte indicates that all bits in the byte are erased.

The erase-verify command must be written to the command register prior to each byte verification to latch its address. The process continues for each byte in the array until a byte does not return FFH data, or the last address is accessed.

In the case where the data read is not FFH, another erase operation is performed. (Refer to Set-up Erase/Erase). Verification then resumes from the address of the last-verified byte. Once all bytes in the array have been verified, the erase step is complete. The device can be programmed. At this point, the verify operation is terminated by writing a valid command (e.g. Program Set-up) to the command register. Figure 5, the Quick-Erase algorithm, illustrates how commands and bus operations are combined to perform electrical erasure of the 28F010. Refer to A.C. Erase Characteristics and Waveforms for specific timing parameters.

Set-up Program/Program Commands

Set-up program is a command-only operation that stages the device for byte programming. Writing 40H into the command register performs the set-up operation.

Once the program set-up operation is performed, the next Write-Enable pulse causes a transition to an active programming operation. Addresses are internally latched on the falling edge of the Write-Enable pulse. Data is internally latched on the rising edge of the Write-Enable pulse. The rising edge of Write-Enable also begins the programming operation. The programming operation terminates with the next rising edge of Write-Enable, used to write the program-verify command. Refer to A.C. Program-

ming Characteristics and Waveforms for specific timing parameters.

Program-Verify Command

The 28F010 is programmed on a byte-by-byte basis. Byte programming may occur sequentially or at random. Following each programming operation, the byte just programmed must be verified.

The program-verify operation is initiated by writing C0H into the command register. The register write terminates the programming operation with the rising edge of its Write-Enable pulse. The program-verify operation stages the device for verification of the byte last programmed. No new address information is latched.

The 28F010 applies an internally-generated margin voltage to the byte. A microprocessor read cycle outputs the data. A successful comparison between the programmed byte and true data means that the byte is successfully programmed. Programming then proceeds to the next desired byte location. Figure 4, the 28F010 Quick-Pulse Programming algorithm, illustrates how commands are combined with bus operations to perform byte programming. Refer to A.C. Programming Characteristics and Waveforms for specific timing parameters.

Reset Command

A reset command is provided as a means to safely abort the erase- or program-command sequences. Following either set-up command (erase or program) with two consecutive writes of FFH will safely abort the operation. Memory contents will not be altered. A valid command must then be written to place the device in the desired state.

EXTENDED ERASE/PROGRAM CYCLING

EEPROM cycling failures have always concerned users. The high electrical field required by thin oxide EEPROMs for tunneling can literally tear apart the oxide at defect regions. To combat this, some suppliers have implemented redundancy schemes, reducing cycling failures to insignificant levels. However, redundancy requires that cell size be doubled—an expensive solution.

Intel has designed extended cycling capability into its ETOX-II flash memory technology. Resulting improvements in cycling reliability come without increasing memory cell size or complexity. First, an advanced tunnel oxide increases the charge carrying ability ten-fold. Second, the oxide area per cell subjected to the tunneling electric field is one-tenth that of common EEPROMs, minimizing the probability of oxide defects in the region. Finally, the peak electric field during erasure is approximately 2 MV/cm lower than EEPROM. The lower electric field greatly reduces oxide stress and the probability of failure—increasing time to wearout by a factor of 100,000,000.

The device is programmed and erased using Intel's Quick-Pulse Programming and Quick-Erase algorithms. Intel's algorithmic approach uses a series of operations (pulses), along with byte verification, to completely and reliably erase and program the device.

QUICK-PULSE PROGRAMMING ALGORITHM

The Quick-Pulse Programming algorithm uses programming operations of 10 μs duration. Each operation is followed by a byte verification to determine when the addressed byte has been successfully programmed. The algorithm allows for up to 25 programming operations per byte, although most bytes verify on the first or second operation. The entire sequence of programming and byte verification is performed with V_{PP} at high voltage. Figure 4 illustrates the Quick-Pulse Programming algorithm.

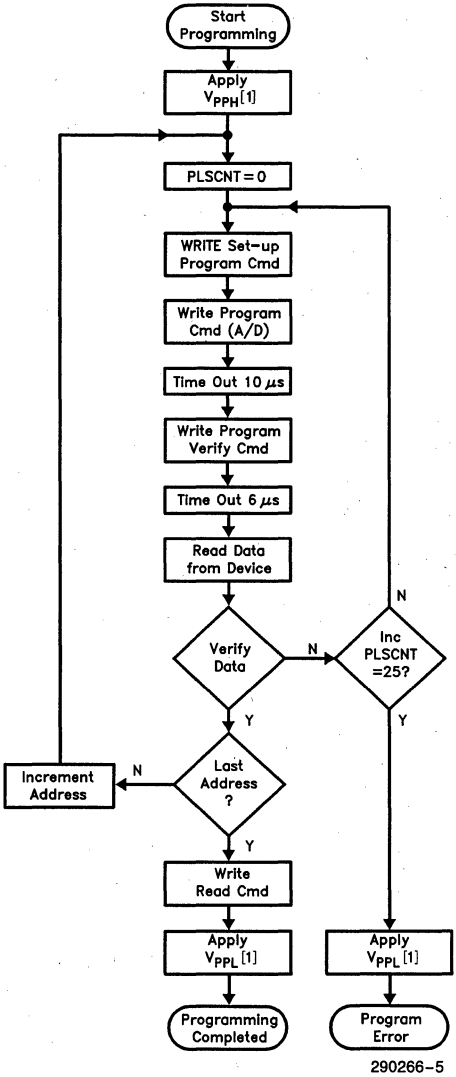
QUICK-ERASE ALGORITHM

Intel's Quick-Erase algorithm yields fast and reliable electrical erasure of memory contents. The algorithm employs a closed-loop flow, similar to the Quick-Pulse Programming algorithm, to simultaneously remove charge from all bits in the array.

Erasure begins with a read of memory contents. The 28F010 is erased when shipped from the factory. Reading FFH data from the device would immediately be followed by device programming.

For devices being erased and reprogrammed, uniform and reliable erasure is ensured by first programming all bits in the device to their charged state (Data = 00H). This is accomplished, using the Quick-Pulse Programming algorithm, in approximately two seconds.

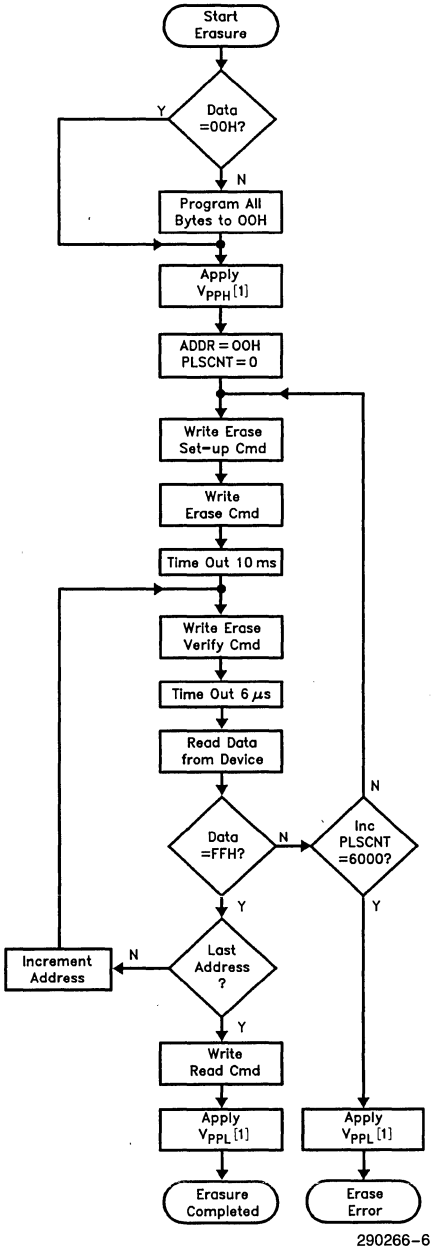
Erase execution then continues with an initial erase operation. Erase verification (data = FFH) begins at address 0000H and continues through the array to the last address, or until data other than FFH is encountered. With each erase operation, an increasing number of bytes verify to the erased state. Erase efficiency may be improved by storing the address of the last byte verified in a register. Following the next erase operation, verification starts at that stored address location. Erasure typically occurs in one second. Figure 5 illustrates the Quick-Erase algorithm.



Bus Operation	Command	Comments
Standby		Wait for V_{PP} Ramp to V_{PPH}(1)
		Initialize Pulse-Count
Write	Set-up Program	Data = 40H
Write	Program	Valid Address/Data
Standby		Duration of Program Operation (t_{WHWH1})
Write	Program(3) Verify	Data = C0H; Stops Program Operation(2)
Standby		t_{WHGL}
Read		Read Byte to Verify Programming
Standby		Compare Data Output to Data Expected
Write	Read	Data = 00H, Resets the Register for Read Operations
Standby		Wait for V_{PP} Ramp to V_{PPL}(1)

NOTES:
1. See DC Characteristics for value of V_{PPH}. The V_{PP} power supply can be hard-wired to the device or switchable. When V_{PP} is switched, V_{PPL} may be ground, no-connect with a resistor tied to ground, or less than 6.5V. Refer to Principles of Operation.
2. Refer to Principles of Operation.
3. Program Verify is only performed after byte programming. A final read/compare may be performed (optional) after the register is written with the Read command.
4. **CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.**

Figure 4. 28F010 Quick-Pulse Programming Algorithm



290266-6

Bus Operation	Command	Comments
		Entire Memory Must = 00H Before Erasure
		Use Quick-Pulse Programming Algorithm (Figure 4)
Standby		Wait for V_{PP} Ramp to V_{PPH}(1)
		Initialize Addresses and Pulse-Count
Write	Set-up Erase	Data = 20H
Write	Erase	Data = 20H
Standby		Duration of Erase Operation (t_{WHWH2})
Write	Erase(3) Verify	Addr = Byte to Verify; Data = A0H; Stops Erase Operation(2)
Standby		t_{WHGL}
Read		Read Byte to Verify Erasure
Standby		Compare Output to FFH Increment Pulse-Count
Write	Read	Data = 00H, Resets the Register for Read Operations
Standby		Wait for V_{PP} Ramp to V_{PPL}(1)

NOTES:

1. See DC Characteristics for value of V_{PPH}. The V_{PP} power supply can be hard-wired to the device or switchable. When V_{PP} is switched, V_{PPL} may be ground, no-connect with a resistor tied to ground, or less than 6.5V. Refer to Principles of Operation.
2. Refer to Principles of Operation.
3. Erase Verify is performed only after chip-erasure. A final read/compare may be performed (optional) after the register is written with the read command.
4. **CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.**

Figure 5. A28F010 Quick-Erase Algorithm

8

DESIGN CONSIDERATIONS

Two-Line Output Control

Flash-memories are often used in larger memory arrays. Intel provides two read-control inputs to accommodate multiple memory connections. Two-line control provides for:

a. the lowest possible memory power dissipation and,

b. complete assurance that output bus contention will not occur.

To efficiently use these two control inputs, an address-decoder output should drive chip-enable, while the system's read signal controls all flash-memories and other parallel memories. This assures that only enabled memory devices have active outputs, while deselected devices maintain the low power standby condition.

Power Supply Decoupling

Flash memory power-switching characteristics require careful device decoupling. System designers are interested in three supply current (I_{CC}) issues—standby, active, and transient current peaks produced by falling and rising edges of chip-enable. The capacitive and inductive loads on the device outputs determine the magnitudes of these peaks.

Two-line control and proper decoupling capacitor selection will suppress transient voltage peaks. Each device should have a 0.1 μF ceramic capacitor connected between V_{CC} and V_{SS}, and between V_{PP} and V_{SS}.

Place the high-frequency, low-inherent-inductance capacitors as close as possible to the devices. Also, for every eight devices, a 4.7 μF electrolytic capacitor should be placed at the array's power supply connection, between V_{CC} and V_{SS}. The bulk capacitor will overcome voltage slumps caused by printed-circuit-board trace inductance, and will supply charge to the smaller capacitors as needed.

V_{PP} Trace on Printed Circuit Boards

Programming flash-memories, while they reside in the target system, requires that the printed circuit board designer pay attention to the V_{PP} power supply trace. The V_{PP} pin supplies the memory cell current for programming. Use similar trace widths and layout considerations given the V_{CC} power bus. Adequate V_{PP} supply traces and decoupling will decrease V_{PP} voltage spikes and overshoots.

Power Up/Down Protection

The 28F010 is designed to offer protection against accidental erasure or programming, caused by spurious system-level signals that may exist during power transitions. Also, with its control register architecture, alteration of memory contents only occurs after successful completion of the two-step command sequences. Power supply sequencing is not required. Internal circuitry of the 28F010 ensures that the command register architecture is reset to the read mode on power up.

A system designer must guard against active writes for V_{CC} voltages above the V_{LKO} when V_{PP} is active. Since both WE# and CE# must be low for a command write, driving either to V_{IH} will prohibit writes. The control register architecture provides an added level of protection since alteration of memory contents only occurs after successful completion of the two-step command sequences.

28F010 Power Dissipation

When designing portable systems, designers must consider battery power consumption not only during device operation, but also for data retention during system idle time. Flash nonvolatility increases the usable battery life of your system because the 28F010 does not consume any power to retain code or data when the system is off. Table 4 illustrates the power dissipated when updating the 28F010.

Table 4. 28F010 Typical Update Power Dissipation(4)

Operation	Power Dissipation (Watt-Seconds)	Notes
Array Program/ Program Verify	0.171	1
Array Erase/ Erase Verify	0.136	2
One Complete Cycle	0.478	3

NOTES:

1. Formula to calculate typical Program/Program Verify Power = [V_{PP} × # Bytes × typical # Prog Pulses (t_{WHWH1} × I_{PP2} typical + t_{WHGL} × I_{PP4} typical)] + [V_{CC} × # Bytes × typical # Prog Pulses (t_{WHWH1} × I_{CC2} typical + t_{WHGL} × I_{CC4} typical].
2. Formula to calculate typical Erase/Erase Verify Power = [V_{PP} (V_{PP3} typical × t_{ERASE} typical + I_{PP5} typical × t_{WHGL} × # Bytes)] + [V_{CC} (I_{CC3} typical × t_{ERASE} typical × I_{CC5} typical × t_{WHGL} × # Bytes)].
3. One Complete Cycle = Array Preprogram + Array Erase + Program.
4. "Typicals" are not guaranteed, but based on a limited number of samples from production lots.

ABSOLUTE MAXIMUM RATINGS*

Operating Temperature
During Read ..... −40°C to +125°C(1)
During Erase/Program ..... −40°C to +125°C

Temperature Under Bias ..... −40°C to +125°C

Storage Temperature ..... −65°C to +150°C

Voltage on Any Pin with
Respect to Ground ..... −2.0V to +7.0V(2)

Voltage on Pin A_9 with
Respect to Ground ..... −2.0V to +13.5V(2, 3)

V_{PP} Supply Voltage with
Respect to Ground
During Erase/Program .... −2.0V to +14.0V(2, 3)

V_{CC} Supply Voltage with
Respect to Ground ..... −2.0V to +7.0V(2)

Output Short Circuit Current ..... 100 mA(4)

Maximum Junction Temperature (T_J) ..... +140°C

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for automotive product defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to −2.0V for periods less than 20 ns. Maximum DC voltage on output pins is V_{CC} + 0.5V, which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.
3. Maximum DC voltage on A_9 or V_{PP} may overshoot to +14.0V for periods less than 20 ns.
4. Output shorted for no more than one second. No more than one output shorted at a time.

OPERATING CONDITIONS

Symbol	Parameter	Limits		Unit	Comments
		Min	Max		
T_A	Operating Temperature	−40	+125	°C	For Read-Only and Read/Write Operations
V_{CC}	V_{CC} Supply Voltage	4.50	5.50	V	

DC CHARACTERISTICS—TTL/NMOS COMPATIBLE

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1			±1.0	µA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1			±10	µA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1			1.0	mA	$V_{CC} = V_{CC}$ Max CE# = V_{IH}
I_{CC1}	V_{CC} Active Read Current	1		10	30	mA	$V_{CC} = V_{CC}$ Max, CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Programming Current	1, 2		1.0	30	mA	Programming in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		5.0	30	mA	Erasure in Progress
I_{CC4}	V_{CC} Program Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±10	µA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1		90	200	µA	$V_{PP} > V_{CC}$
					±10	µA	$V_{PP} \leq V_{CC}$

8

DC CHARACTERISTICS—TTL/NMOS COMPATIBLE (Continued)

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{PP2}	V_{PP} Programming Current	1, 2		8.0	30	mA	$V_{PP} = V_{PPH}$ Programming in Progress
I_{PP3}	V_{PP} Erase Current	1, 2		4.0	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Program Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.0		$V_{CC} + 0.5$	V	
V_{OL}	Output Low Voltage				0.45	V	$I_{OL} = 2.1$ mA $V_{CC} = V_{CC}$ Min
V_{OH1}	Output High Voltage		2.4			V	$I_{OH} = -2.5$ mA $V_{CC} = V_{CC}$ Min
V_{ID}	A_9 Intelligent Identifer Voltage		11.50		13.00	V	$A_9 = V_{ID}$
I_{ID}	V_{CC} ID Current	1		10	30	mA	$A_9 = V_{ID}$
	V_{PP} ID CURRENT			90	500	μA	
V_{PPL}	V_{PP} during Read-Only Operations		0.00		6.5	V	**NOTE:** Erase/Program are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} during Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

DC CHARACTERISTICS—CMOS COMPATIBLE

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1			±1.0	μA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1			±10	μA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1			300	μA	$V_{CC} = V_{CC}$ Max CE# = V_{CC} ±0.2V
I_{CC1}	V_{CC} Active Read Current	1		10	30	mA	$V_{CC} = V_{CC}$ Max, $\overline{CE} = V_{IL}$ f = 6 MHz, $I_{OUT} = 0$ mA
I_{CC2}	V_{CC} Programming Current	1, 2		1.0	30	mA	Programming in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		5.0	30	mA	Erasure in Progress
I_{PPS}	V_{PP} Leakage Current	1			±10	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1		90	200	μA	$V_{PP} > V_{CC}$
					±10		$V_{PP} \leq V_{CC}$

DC CHARACTERISTICS—CMOS COMPATIBLE (Continued)

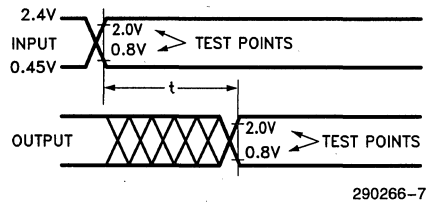
Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{PP2}	V_{PP} Programming Current	1, 2		8.0	30	mA	$V_{PP} = V_{PPH}$ Programming in Progress
I_{PP3}	V_{PP} Erase Current	1, 2		4.0	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Program Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 2		5.0	5.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage	−0.5			0.8	V	
V_{IH}	Input High Voltage	0.7 V_{CC}			V_{CC} + 0.5	V	
V_{OL}	Output Low Voltage				0.45	V	I_{OL} = 2.1 mA $V_{CC} = V_{CC}$ Min
V_{OH1}	Output High Voltage	0.85 V_{CC}				V	I_{OH} = −2.5 mA, $V_{CC} = V_{CC}$ Min
V_{OH2}		V_{CC} − 0.4					I_{OH} = −100 μA, $V_{CC} = V_{CC}$ Min
V_{ID}	A_9 Intelligent Identifier Voltage		11.50		13.00	V	
I_{ID}	V_{CC} ID Current	1		10	30	mA	A_9 = ID
I_{ID}	V_{PP} ID Current	1		90	500	μA	A_9 = ID
V_{PPL}	V_{PP} during Read-Only Operations	0.00			6.5	V	**NOTE:** Erase/Programs are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} during Read/Write Operations	11.40			12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage	2.5				V	

CAPACITANCE[3] T_A = 25°C, f = 1.0 MHz

Symbol	Parameter	Limits		Unit	Conditions
		Min	Max		
C_{IN}	Address/Control Capacitance		8	pF	V_{IN} = 0V
C_{OUT}	Output Capacitance		12	pF	V_{OUT} = 0V

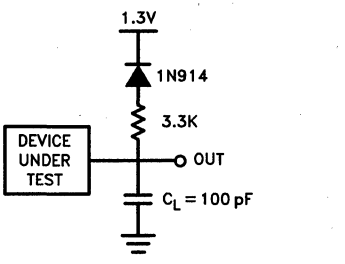
NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T= 25°C.
2. Not 100% tested: characterization data available.
3. Sampled, not 100% tested.
4. "Typicals" are not guaranteed, but are based on a limited number of samples from production lots.

AC TESTING INPUT/OUTPUT WAVEFORM



AC Testing: Inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0". Testing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0". Rise/Fall time ≤ 10 ns.

AC TESTING LOAD CIRCUIT



C_L = 100 pF
C_L includes Jig Capacitance

AC TEST CONDITIONS

Input Rise and Fall Times (10% to 90%) . . . . . 10 ns
Input Pulse Levels . . . . . 0.45V and 2.4V
Input Timing Reference Level . . . . . 0.8V and 2.0V
Output Timing Reference Level . . . . . 0.8V and 2.0V

AC CHARACTERISTICS—Read-Only Operations(2)

Versions		Notes	28F010-120		28F010-150		Unit
Symbol	Characteristic		Min	Max	Min	Max	
t_{AVAV}/t_{RC}	Read Cycle Time	3	120		150		ns
t_{ELQV}/t_{CE}	Chip Enable Access Time			120		150	ns
t_{AVQV}/t_{ACC}	Address Access Time			120		150	ns
t_{GLQV}/t_{OE}	Output Enable Access Time			50		55	ns
t_{ELQX}/t_{LZ}	Chip Enable to Output in Low Z	3	0		0		ns
t_{EHQZ}	Chip Disable to Output in High Z	3		55		55	ns
t_{GLQX}/t_{OLZ}	Output Enable to Output in Low Z	3	0		0		ns
t_{GHQZ}/t_{DF}	Output Disable to Output in High Z	4		30		35	ns
t_{OH}	Output Hold from Address, CE#, or OE# Change	1, 3	0		0		ns
t_{WHGL}	Write Recovery Time before Read		6		6		μs

NOTES:
1. Whichever occurs first.
2. Rise/Fall Time ≤ 10 ns.
3. Not 100% tested: characterization data available.
4. Guaranteed by design.

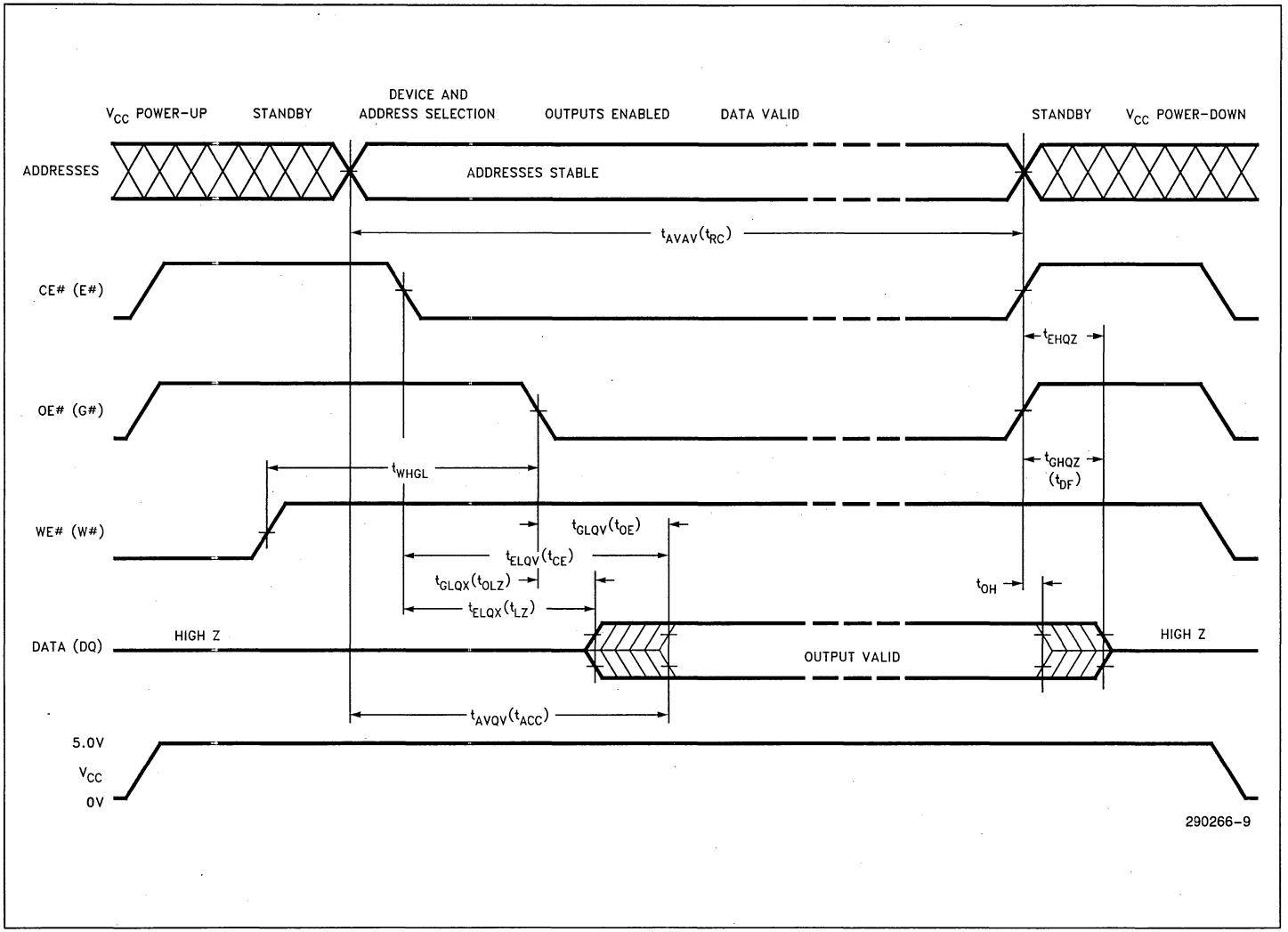


Figure 6. AC Waveforms for Read Operations

AC CHARACTERISTICS—Write/Erase/Program Operations[1, 3]

Versions		Notes	28F010-120		28F010-150		Unit
Symbol	Characteristic		Min	Max	Min	Max	
t_{AVAV}/t_{WC}	Write Cycle Time		120		150		ns
t_{AVWL}/t_{AS}	Address Set-Up Time		0		0		ns
t_{WLAX}/t_{AH}	Address Hold Time	2	60		60		ns
t_{DVWH}/t_{DS}	Data Set-up Time		50		50		ns
t_{WHDX}/t_{DH}	Data Hold Time		10		10		ns
t_{WHGL}	Write Recovery Time before Read		6		6		μs
t_{GHWL}	Read Recovery Time before Write		0		0		μs
t_{ELWL}/t_{CS}	Chip Enable Set-Up Time before Write	2	20		20		ns
t_{WHEH}/t_{CH}	Chip Enable Hold Time		0		0		ns
t_{WLWH}/t_{WP}	Write Pulse Width[2]	2	80		80		ns
t_{ELEH}	Alternative Write[2] Pulse Width	2	80		80		ns
t_{WHWL}/t_{WPH}	Write Pulse Width High		20		20		ns
t_{WHWH1}	Duration of Programming Operation	4	10		10		μs
t_{WHWH2}	Duration of Erase Operation	4	9.5		9.5		ms
t_{VPEL}	V_{PP} Set-Up Time to Chip Enable Low		1.0		1.0		ms

NOTES:
1. Read timing characteristics during read/write operations are the same as during read-only operations. Refer to AC Characteristics for Read-Only Operations.
2. Chip-Enable Controlled Writes: Write operations are driven by the valid combination of Chip-Enable and Write-Enable. In systems where Chip-Enable defines the write pulse width (within a longer Write-Enable timing waveform) all set-up, hold, and inactive Write-Enable times should be measured relative to the Chip-Enable waveform.
3. Rise/Fall time ≤ 10 ns.
4. The internal stop timer terminates the programming/erase operations, thereby eliminating the need for a maximum specification.

ERASE AND PROGRAMMING PERFORMANCE

Parameter	Notes	Limits			Unit	Comments
		Min	Typ	Max		
Chip Erase Time	1, 3, 4, 6		1	60	Sec	Excludes 00H Programming Prior to Erasure
Chip Program Time	1, 2, 4		2	12.5	Sec	Excludes System-Level Overhead
Erase/Program Cycles	1, 5	1,000	100,000		Cycles	

NOTES:
1. "Typicals" are not guaranteed, but based on a limited number of samples taken from production lots. Data taken at $T = 25°C$, $V_{PP} = 12.0V$, $V_{CC} = 5.0V$.
2. Minimum byte programming time excluding system overhead is 16 μsec (10 μsec program + 6 μsec write recovery), while maximum is 400 μsec/byte (16 μsec x 25 loops allowed by algorithm). Max chip programming time is specified lower than the worst case allowed by the programming algorithm since most bytes program significantly faster than the worst case byte.
3. Excludes 00H programming prior to erasure.
4. Excludes system-level overhead.
5. Refer to RR-60 "ETOX Flash Memory Reliability Data Summary" for typical cycling data and failure rate calculations.
6. Maximum erase specification is determined by algorithmic limit and accounts for cumulative effect of erasure at $T = -40°C$, 1,000 cycles, $V_{PP} = 11.4V$, $V_{CC} = 4.5V$.

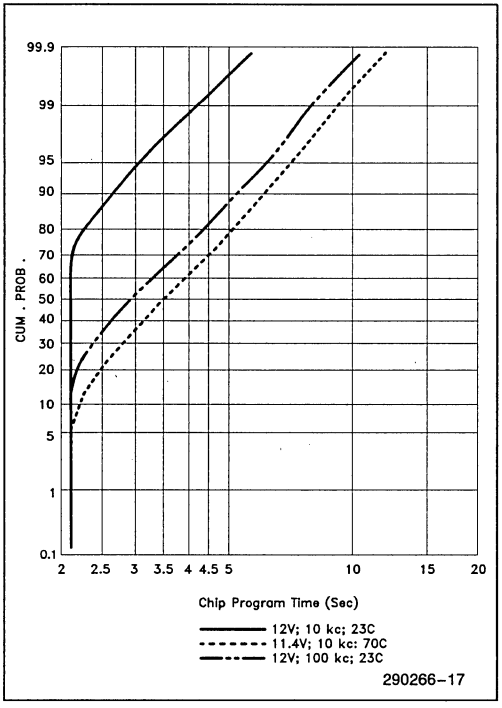


Figure 7. 28F010 Typical Programming Capability
See Note 1, Page 18.

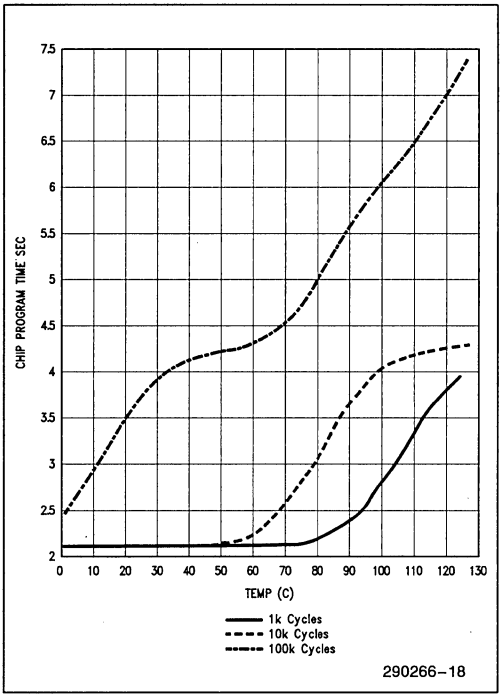


Figure 8. 28F010 Typical Program Time at 12V

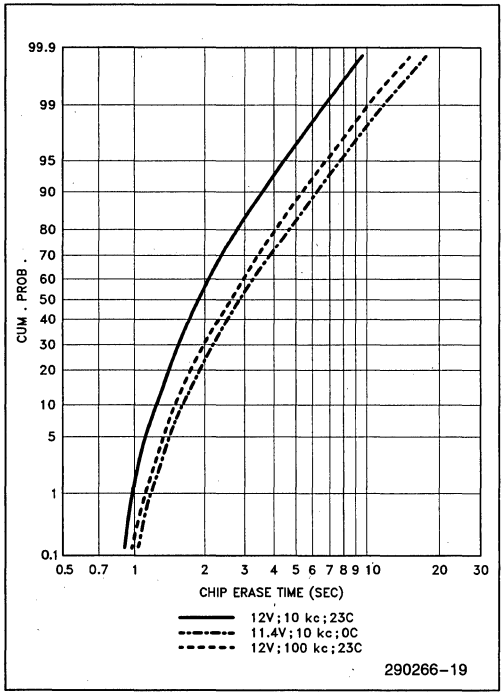


Figure 9. 28F010 Typical Erase Capability
See Note 1, Page 18.

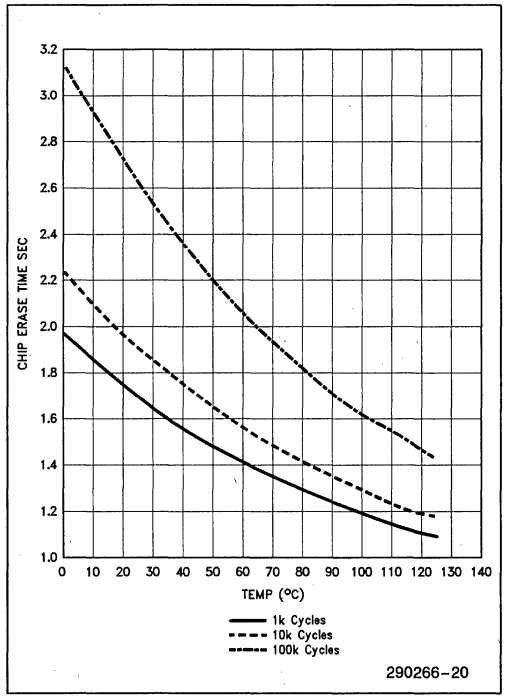


Figure 10. 28F010 Typical Erase Time at 12V

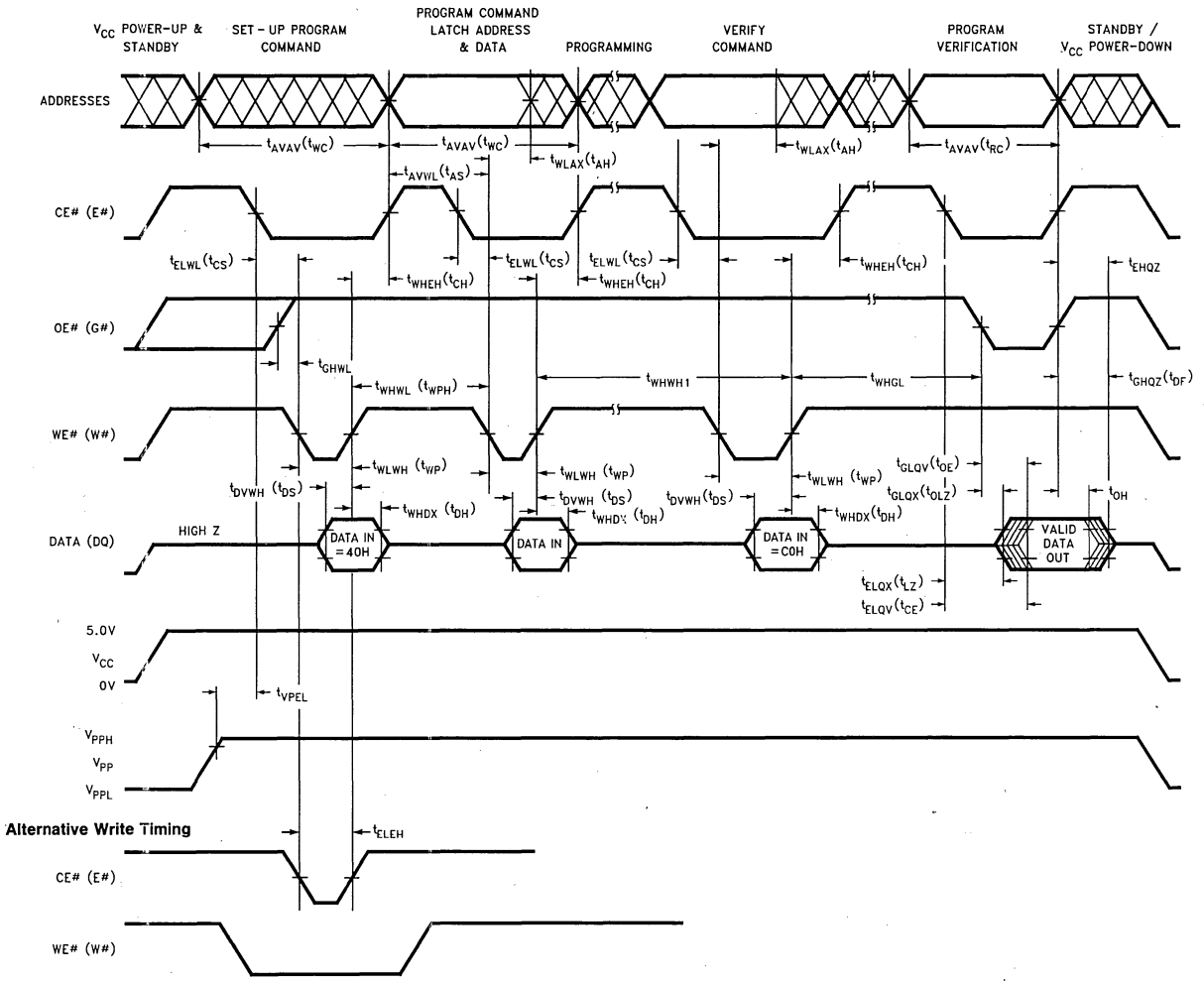


Figure 11. AC Waveforms for Programming Operations

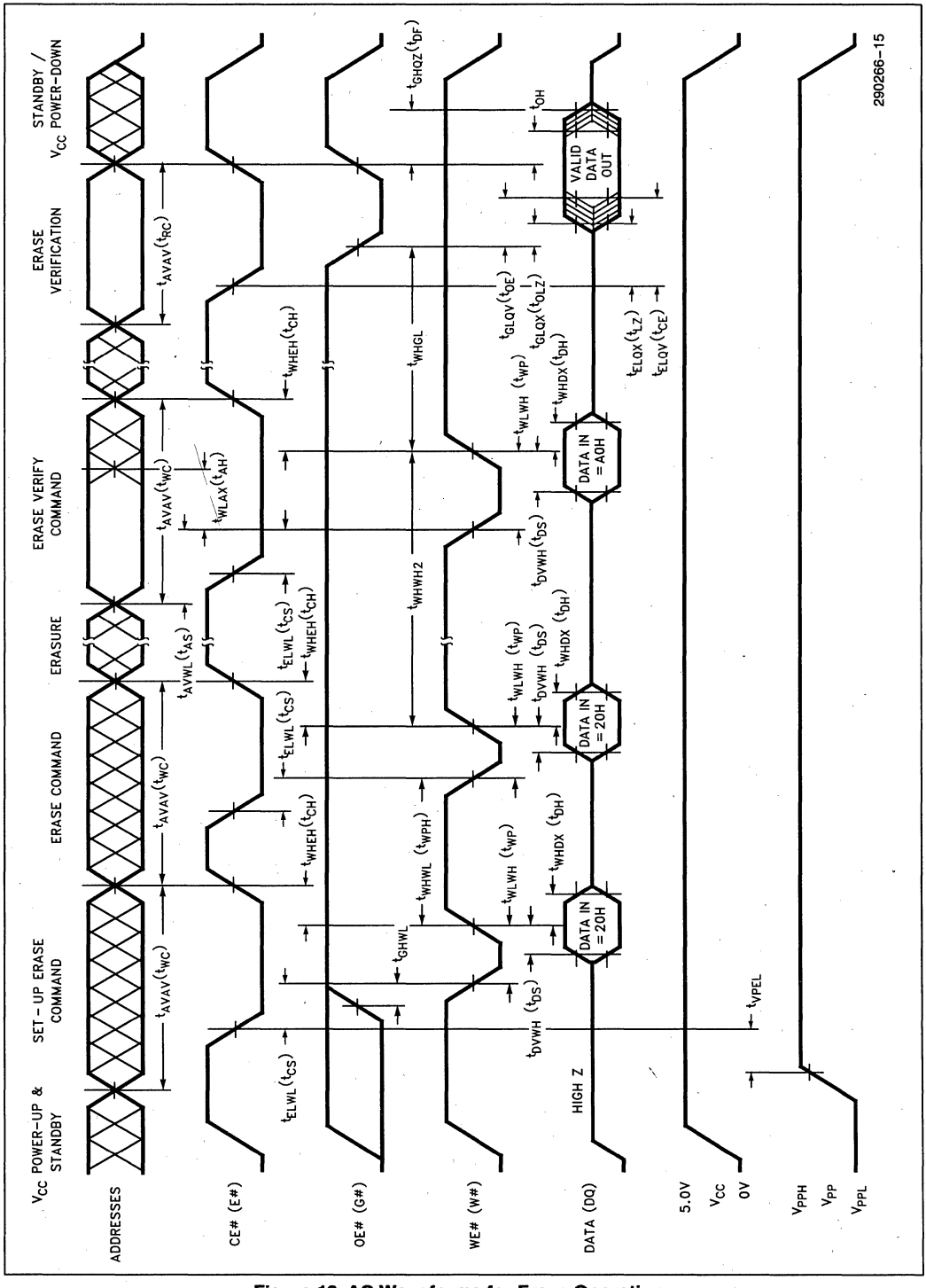
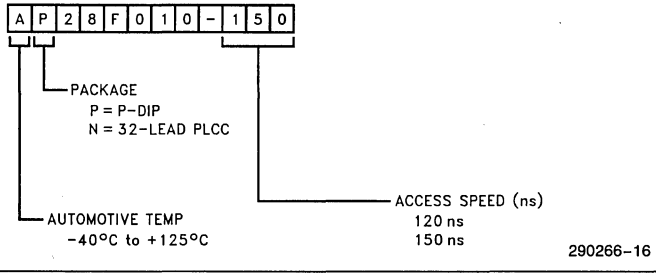


Figure 12. AC Waveforms for Erase Operations

Ordering Information



Valid Combinations:

AP28F010-120	AN28F010-120
AP28F010-150	AN28F010-150

ADDITIONAL INFORMATION

	Order Number
ER-20, "ETOX™ II Flash Memory Technology"	294005
ER-24, "Intel Flash Memory"	294008
RR-60, "ETOX™ Flash Memory Reliability Data Summary"	293002
AP-316, "Using Flash Memory for In-System Reprogrammable Nonvolatile Storage"	292046
AP-325, "Guide to Flash Memory Reprogramming"	292059

REVISION HISTORY

Number	Description
003	Changed Erase/Program Cycles to 1,000 minimum
004	Added 120 ns Characteristics

8

A28F512
512K (64K x 8) CMOS FLASH MEMORY
(Automotive)

- **Extended Automotive Temperature Range: −40°C to +125°C**
- **Flash Electrical Chip-Erase**
  - **1 Second Typical Chip-Erase**
- **Quick-Pulse Programming Algorithm**
  - **10 μs Typical Byte-Program**
  - **1 Second Chip-Program**
- **1,000 Erase/Program Cycle Minimum Over Automotive Temperature Range**
- **12.0V ±5% V_{PP}**
- **High-Performance Read**
  - **120/150 ns Maximum Access Time**
- **CMOS Low Power Consumption**
  - **30 mA Maximum Active Current**
  - **100 μA Maximum Standby Current**
- **Integrated Program/Erase Stop Timer**
- **Command Register Architecture for Microprocessor/Microcontroller Compatible Write Interface**
- **Noise Immunity Features**
  - **± 10% V_{CC} Tolerance**
  - **Maximum Latch-Up Immunity through EPI Processing**
- **ETOX™ II Flash Nonvolatile Memory Technology**
  - **EPROM-Compatible Process Base**
  - **High-Volume Manufacturing Experience**
- **JEDEC-Standard Pinouts**
  - **32-Pin Plastic DIP**
  - **32-Pin PLCC**

  (See Packaging Spec., Order #231369)

Intel's 28F512 CMOS flash memory offers the most cost-effective and reliable alternative for read/write random access nonvolatile memory. The 28F512 adds electrical chip-erasure and reprogramming to familiar EPROM technology. Memory contents can be rewritten: in a test socket; in a PROM-programmer socket; on-board during subassembly test; in-system during final test; and in-system after-sale. The 28F512 increases memory flexibility, while contributing to time- and cost-savings.

The 28F512 is a 512-kilobit nonvolatile memory organized as 65,536 bytes of 8 bits. Intel's 28F512 is offered in 32-pin cerdip or 32-lead PLCC packages. Pin assignments conform to JEDEC standards.

Extended erase and program cycling capability is designed into Intel's ETOX™ II (EPROM Tunnel Oxide) process technology. Advanced oxide processing, an optimized tunneling structure, and lower electric field combine to extend reliable cycling beyond that of traditional EEPROMs. With the 12.0V V_{PP} supply, the 28F512 performs a minimum of 1,000 erase and program cycles well within the time limits of the Quick-Pulse Programming and Quick-Erase algorithms.

Intel's 28F512 employs advanced CMOS circuitry for systems requiring high-performance access speeds, low power consumption, and immunity to noise. Its 150 nanosecond access time provides no-WAIT-state performance for a wide range of microprocessors and microcontrollers. Maximum standby current of 100 μA translates into power savings when the device is deselected. Finally, the highest degree of latch-up protection is achieved through Intel's unique EPI processing. Prevention of latch-up is provided for stresses up to 100 mA on address and data pins, from −1V to V_{CC} + 1V.

With Intel's ETOX II process base, the 28F512 leverages years of EPROM experience to yield the highest levels of quality, reliability, and cost-effectiveness.

In order to meet the rigorous environmental requirements of Automotive applications, Intel offers the 28F512 in extended Automotive temperature range. Read and Write Characteristics are guaranteed over the range of −40°C to +125°C ambient.

March 1994
Order Number: 290265-004

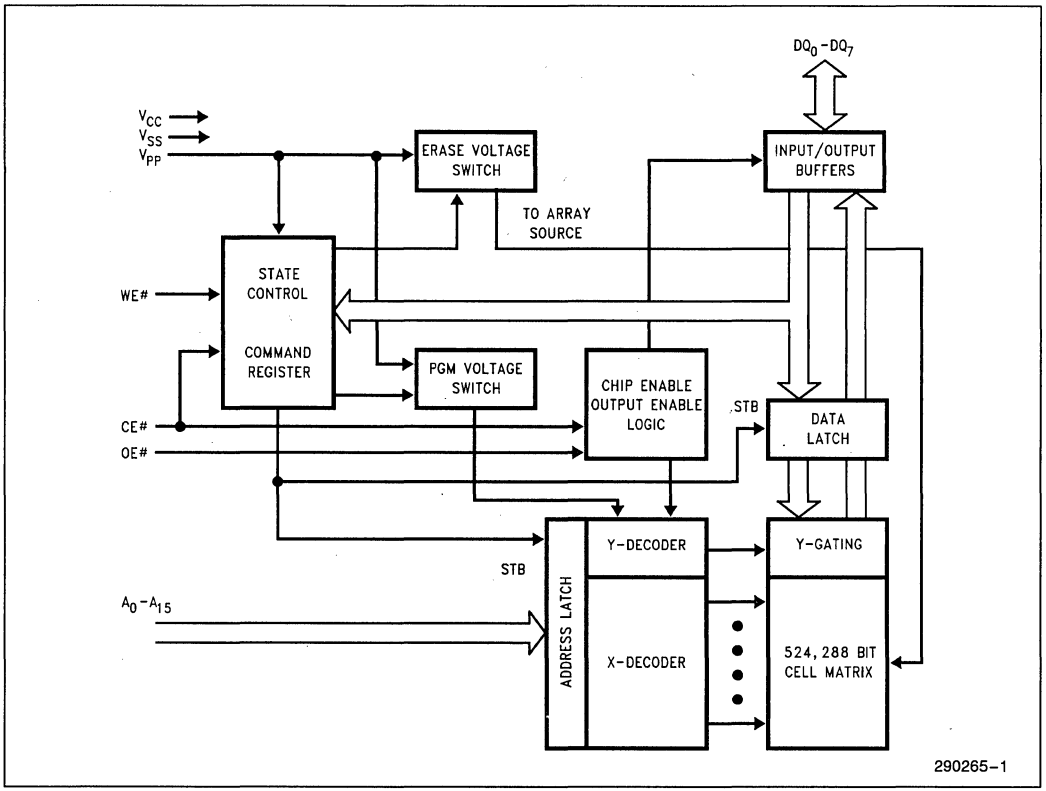


Figure 1. 28F512 Block Diagram

AUTOMOTIVE TEMPERATURE FLASH MEMORIES

The Intel Automotive Flash Memories have received additional processing to enhance product characteristics. The automotive temperature range is −40°C to +125°C during the read/write/erase/program operations.

Speed Versions	Packaging Options	
	Plastic DIP	PLCC
-120	AP	AN
-150	AP	AN

8

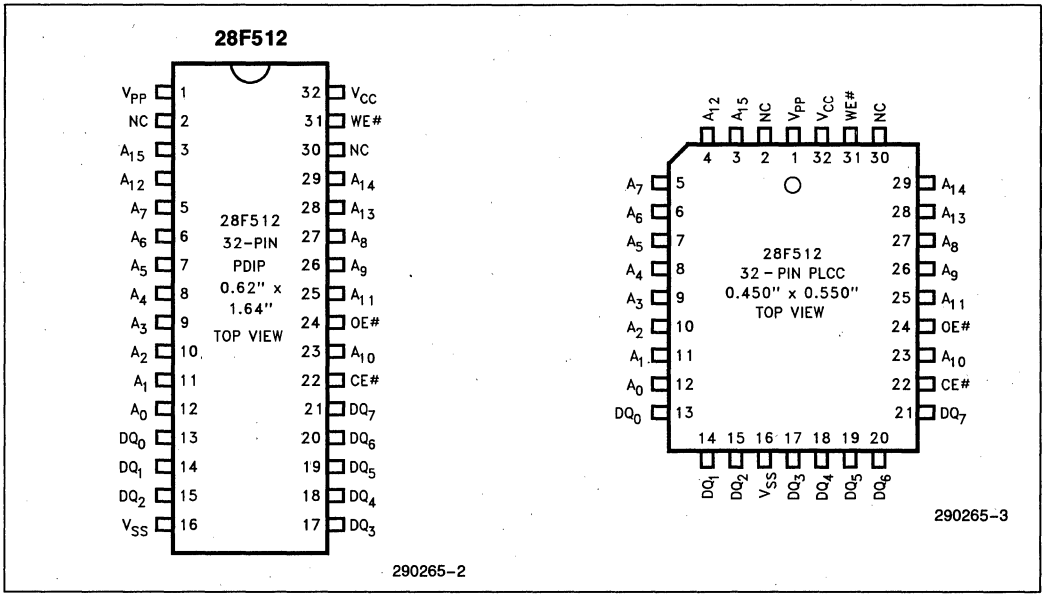


Figure 2. 28F512 Pin Configurations

Table 1. Pin Description

Symbol	Type	Name and Function
A_0–A_{15}	INPUT	**ADDRESS INPUTS** for memory addresses. Addresses are internally latched during a write cycle.
DQ_0–DQ_7	INPUT/OUTPUT	**DATA INPUT/OUTPUT:** Inputs data during memory write cycles; outputs data during memory read cycles. The data pins are active high and float to tri-state OFF when the chip is deselected or the outputs are disabled. Data is internally latched during a write cycle.
CE#	INPUT	**CHIP ENABLE:** Activates the device's control logic, input buffers, decoders and sense amplifiers. CE# is active low; CE# high deselects the memory device and reduces power consumption to standby levels.
OE#	INPUT	**OUTPUT ENABLE:** Gates the devices output through the data buffers during a read cycle. OE# is active low.
WE#	INPUT	**WRITE ENABLE:** Controls writes to the control register and the array. Write enable is active low. Addresses are latched on the falling edge and data is latched on the rising edge of the WE# pulse. **Note:** With $V_{PP} \leq 6.5V$, memory contents cannot be altered.
V_{PP}		**ERASE/PROGRAM POWER SUPPLY** for writing the command register, erasing the entire array, or programming bytes in the array.
V_{CC}		**DEVICE POWER SUPPLY** (5V ±10%)
V_{SS}		**GROUND**
NC		**NO INTERNAL CONNECTION** to device. Pin may be driven or left floating.

APPLICATIONS

The 28F512 flash-memory adds electrical chip-erasure and reprogrammability to EPROM non-volatility and ease of use. The 28F512 is ideal for storing code or data-tables in applications where periodic updates are required. The 28F512 also serves as a dense, nonvolatile data acquisition and storage medium.

The need for code updates pervades all phases of a system's life—from prototyping to system manufacture to after-sale service. In the factory, during prototyping, revisions to control code necessitate ultraviolet erasure and reprogramming of EPROM-based prototype codes. The 28F512 replaces the 15- to 20-minute ultraviolet erasure with one-second electrical erasure. Electrical chip-erasure and reprogramming occur in the same workstation or PROM-programmer socket.

Diagnostics, performed at subassembly or final assembly stages, often require the socketing of EPROMs. Socketed test codes are ultimately replaced with EPROMs containing the final program. With electrical chip-erasure and reprogramming, the 28F512 is soldered to the circuit board. Test codes are programmed into the 28F512 as it resides on the circuit board. Ultimately, the final code can be downloaded to the device. The 28F512's in-circuit alterability eliminates unnecessary handling and less-reliable socketed connections, while adding greater test flexibility.

Material and labor costs associated with code changes increase at higher levels of system integration—the most costly being code updates after sale. Code "bugs", or the desire to augment system functionality, prompt after-sale code updates. Field revisions to EPROM-based code require the removal of EPROM components or entire boards.

Designing with the in-circuit alterable 28F512 eliminates socketed memories, reduces overall material costs, and drastically cuts the labor costs associated with code updates. With the 28F512, code updates are implemented locally via an edge-connector, or remotely over a serial communication link.

The 28F512's electrical chip-erasure, byte reprogrammability, and complete nonvolatility fit well with data accumulation needs. Electrical chip-erasure gives the designer a "blank-slate" in which to log data. Data can be periodically off-loaded for analysis—erasing the slate and repeating the cycle. Or, multiple devices can maintain a "rolling window" of accumulated data.

With high density, nonvolatility, and extended cycling capability, the 28F512 offers an innovative alternative for mass storage.

Integrating main memory and backup storage functions into directly executable flash memory boosts system performance, shrinks system size, and cuts power consumption. Reliability exceeds that of electromechanical media, with greater durability in extreme environmental conditions.

A high degree of on-chip feature integration simplifies memory-to-processor interfacing. Figure 3 depicts two 28F512s tied to the 80C186 system bus. The 28F512's architecture minimizes interface circuitry needed for complete in-circuit updates of memory contents.

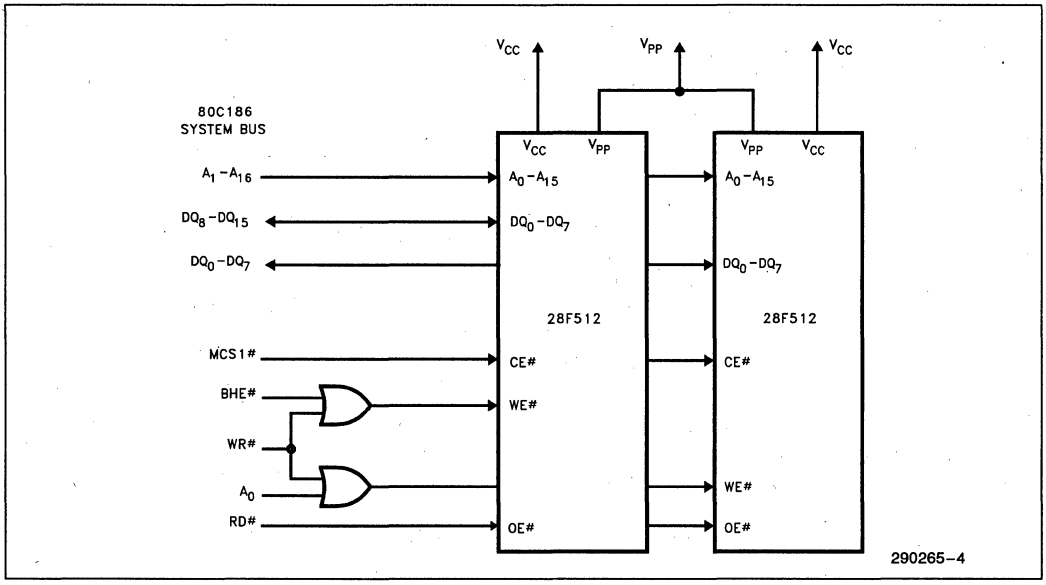


Figure 3. 28F512 in an 80C186 System

With cost-effective in-system reprogramming and extended cycling capability, the 28F512 fills the functionality gap between traditional EPROMs and EEPROMs. EPROM-compatible specifications, straightforward interfacing, and in-circuit alterability allows designers to easily augment memory flexibility and satisfy the need for updatable nonvolatile storage in today's designs.

PRINCIPLES OF OPERATION

Flash-memory augments EPROM functionality with in-circuit electrical erasure and reprogramming. The 28F512 introduces a command register to manage this new functionality. The command register allows for: 100% TTL-level control inputs; fixed power supplies during erasure and programming; and maximum EPROM compatibility.

In the absence of high voltage on the V_{PP} pin, the 28F512 is a read-only memory. Manipulation of the external memory-control pins yields the standard EPROM read, standby, output disable, and Intelligent Identifier operations.

The same EPROM read, standby, and output disable operations are available when high voltage is applied to the V_{PP} pin. In addition, high voltage on V_{PP} enables erasure and programming of the device. All functions associated with altering memory contents—Intelligent Identifier, erase, erase verify, program, and program verify—are accessed via the command register.

Commands are written to the register using standard microprocessor write timings. Register contents serve as input to an internal state-machine which controls the erase and programming circuitry. Write cycles also internally latch addresses and data needed for programming or erase operations. With the appropriate command written to the register, standard microprocessor read timings output array data, access the Intelligent Identifier codes, or output data for erase and program verification.

Integrated Program/Erase Stop Timer

Successive command write cycles define the durations of program and erase operations; specifically, the program or erase time durations are normally terminated by associated program or erase verify commands. An integrated stop timer provides simplified timing control over these operations; thus eliminating the need for maximum program/erase timing specifications. Program and erase pulse durations are minimums only. When the stop timer terminates a program or erase operation, the device enters an inactive state and remains inactive until receiving the appropriate verify or reset command.

Write Protection

The command register is only alterable when V_{PP} is at high voltage. Depending upon the application, the system designer may choose to make the V_{PP} power supply switchable—available only when memory updates are desired. When high voltage is removed,

Table 2. 28F512 Bus Operations

Operation	Pins	V_{PP}(1)	A_0	A_9	CE#	OE#	WE#	DQ_0–DQ_7
READ-ONLY	Read	V_{PPL}	A_0	A_9	V_{IL}	V_{IL}	V_{IH}	Data Out
	Output Disable	V_{PPL}	X	X	V_{IL}	V_{IH}	V_{IH}	Tri-State
	Standby	V_{PPL}	X	X	V_{IH}	X	X	Tri-State
	Intelligent Identifier (Mfr)(2)	V_{PPL}	V_{IL}	V_{ID}(3)	V_{IL}	V_{IL}	V_{IH}	Data = 89H
	Intelligent Identifier (Device)(2)	V_{PPL}	V_{IH}	V_{ID}(3)	V_{IL}	V_{IL}	V_{IH}	Data = B8H
READ/WRITE	Read	V_{PPH}	A_0	A_9	V_{IL}	V_{IL}	V_{IH}	Data Out(4)
	Output Disable	V_{PPH}	X	X	V_{IL}	V_{IH}	V_{IH}	Tri-State
	Standby(5)	V_{PPH}	X	X	V_{IH}	X	X	Tri-State
	Write	V_{PPH}	A_0	A_9	V_{IL}	V_{IH}	V_{IL}	Data In(6)

NOTES:
1. V_{PPL} may be ground, a no-connect with a resistor tied to ground, or ≤ 6.5V. V_{PPH} is the programming voltage specified for the device. Refer to D.C. Characteristics. When $V_{PP} = V_{PPL}$ memory contents can be read but not written or erased.
2. Manufacturer and device codes may also be accessed via a command register write sequence. Refer to Table 3. All other addresses low.
3. V_{ID} is the Intelligent Identifier high voltage. Refer to DC Characteristics.
4. Read operations with $V_{PP} = V_{PPH}$ may access array data or the Intelligent Identifier codes.
5. With V_{PP} at high voltage, the standby current equals $I_{CC} + I_{PP}$ (standby).
6. Refer to Table 3 for valid Data-In during a write operation.
7. X can be V_{IL} or V_{IH}.

the contents of the register default to the read command, making the 28F512 a read-only memory. Memory contents cannot be altered.

Or, the system designer may choose to "hardwire" V_{PP}, making the high voltage supply constantly available. In this instance, all operations are performed in conjunction with the command register. The 28F512 is designed to accommodate either design practice, and to encourage optimization of the processor-memory interface.

The two-step program/erase write sequence to the Command Register provides additional software write protection.

BUS OPERATIONS

Read

The 28F512 has two control functions, both of which must be logically active, to obtain data at the outputs. Chip-Enable (CE#) is the power control and should be used for device selection. Output-Enable (OE#) is the output control and should be used to gate data from the output pins, independent of device selection. Figure 6 illustrates read timing waveforms.

When V_{PP} is low (V_{PPL}), the read only operation is active. This permits reading the data in the array and outputting the Intelligent Identifier codes (see Table 2). When V_{PP} is high (V_{PPH}), the default condition of the device is the read only mode. This allows reading the data in the array. Further functionality is achieved though the Command Register as shown in Table 3.

Output Disable

With Output-Enable at a logic-high level (V_{IH}), output from the device is disabled. Output pins are placed in a high-impedance state.

Standby

With Chip-Enable at a logic-high level, the standby operation disables most of the 28F512's circuitry and substantially reduces device power consumption. The outputs are placed in a high-impedance state, independent of the Output-Enable signal. If the 28F512 is deselected during erasure, programming, or program/erase verification, the device draws active current until the operation is terminated.

Intelligent Identifier Operation

The Intelligent Identifier operation outputs the manufacturer code (89H) and device code (B8H). Pro-

8

gramming equipment automatically matches the device with its proper erase and programming algorithms.

With Chip-Enable and Output-Enable at a logic low level, raising A9 to high voltage V_{ID} (see DC Characteristics) activates the operation. Data read from locations 0000H and 0001H represent the manufacturer's code and the device code, respectively.

The manufacturer- and device-codes can also be read via the command register, for instances where the 28F512 is erased and reprogrammed in the target system. Following a write of 90H to the command register, a read from address location 0000H outputs the manufacturer code (89H). A read from address 0001H outputs the device code (B8H).

Write

Device erasure and programming are accomplished via the command register, when high voltage is applied to the V_{PP} pin. The contents of the register serve as input to the internal state-machine. The state-machine outputs dictate the function of the device.

The command register itself does not occupy an addressable memory location. The register is a latch used to store the command, along with address and data information needed to execute the command.

The command register is written by bringing Write-Enable to a logic-low level (V_{IL}), while Chip-Enable is low. Addresses are latched on the falling edge of Write-Enable, while data is latched on the rising edge of the Write-Enable pulse. Standard microprocessor write timings are used.

Refer to A.C. Write Characteristics and the Erase/Programming Waveforms for specific timing parameters.

COMMAND DEFINITIONS

When low voltage is applied to the V_{PP} pin, the contents of the command register default to 00H, enabling read-only operations.

Placing high voltage on the V_{PP} pin enables read/write operations. Device operations are selected by writing specific data patterns into the command register. Table 3 defines these 28F512 register commands.

Table 3. Command Definitions

Command	Bus Cycles Req'd	First Bus Cycle			Second Bus Cycle		
		Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory	1	Write	X	00H			
Read Intelligent Identifier Code(4)	2	Write	X	90H	Read	IA	ID
Set-up Erase/Erase(5)	2	Write	X	20H	Write	X	20H
Erase Verify(5)	2	Write	EA	A0H	Read	X	EVD
Set-up Program/Program(6)	2	Write	X	40H	Write	PA	PD
Program Verify(6)	2	Write	X	C0H	Read	X	PVD
Reset(7)	2	Write	X	FFH	Write	X	FFH

NOTES:
1. Bus operations are defined in Table 2.
2. IA = Identifier address: 00H for manufacturer code, 01H for device code.
   EA = Address of memory location to be read during erase verify.
   PA = Address of memory location to be programmed.
   Addresses are latched on the falling edge of the Write-Enable pulse.
3. ID = Data read from location IA during device identification (Mfr = 89H, Device = B8H).
   EVD = Data read from location EA during erase verify.
   PD = Data to be programmed at location PA. Data is latched on the rising edge of Write-Enable.
   PVD = Data read from location PA during program verify. PA is latched on the Program command.
4. Following the Read Intelligent ID command, two read operations access manufacturer and device codes.
5. Figure 5 illustrates the Quick-Erase algorithm.
6. Figure 4 illustrates the Quick-Pulse Programming algorithm.
7. The second bus cycle must be followed by the desired command register write.

Read Command

While V_{PP} is high, for erasure and programming, memory contents can be accessed via the read command. The read operation is initiated by writing 00H into the command register. Microprocessor read cycles retrieve array data. The device remains enabled for reads until the command register contents are altered.

The default contents of the register upon V_{PP} power-up is 00H. This default value ensures that no spurious alteration of memory contents occurs during the V_{PP} power transition. Where the V_{PP} supply is hard-wired to the 28F512, the device powers-up and remains enabled for reads until the command-register contents are changed. Refer to the A.C. Read Characteristics and Waveforms for specific timing parameters.

Intelligent Identifier Command

Flash-memories are intended for use in applications where the local CPU alters memory contents. As such, manufacturer- and device-codes must be accessible while the device resides in the target system. PROM programmers typically access signature codes by raising A9 to a high voltage. However, multiplexing high voltage onto address lines is not a desired system-design practice.

The 28F512 contains an Intelligent Identifier operation to supplement traditional PROM-programming methodology. The operation is initiated by writing 90H into the command register. Following the command write, a read cycle from address 0000H retrieves the manufacturer code of 89H. A read cycle from address 0001H returns the device code of B8H. To terminate the operation, it is necessary to write another valid command into the register.

Set-up Erase/Erase Commands

Set-up Erase is a command-only operation that stages the device for electrical erasure of all bytes in the array. The set-up erase operation is performed by writing 20H to the command register.

To commence chip-erasure, the erase command (20H) must again be written to the register. The erase operation begins with the rising edge of the Write-Enable pulse and terminates with the rising edge of the next Write-Enable pulse (i.e., Erase-Verify Command).

This two-step sequence of set-up followed by execution ensures that memory contents are not accidentally erased. Also, chip-erasure can only occur when high voltage is applied to the V_{PP} pin. In the absence of this high voltage, memory contents are protected against erasure. Refer to A.C. Erase Characteristics and Waveforms for specific timing parameters.

Erase-Verify Command

The erase command erases all bytes of the array in parallel. After each erase operation, all bytes must be verified. The erase verify operation is initiated by writing A0H into the command register. The address for the byte to be verified must be supplied as it is latched on the falling edge of the Write-Enable pulse. The register write terminates the erase operation with the rising edge of its Write-Enable pulse.

The 28F512 applies an internally-generated margin voltage to the addressed byte. Reading FFH from the addressed byte indicates that all bits in the byte are erased.

The erase-verify command must be written to the command register prior to each byte verification to latch its address. The process continues for each byte in the array until a byte does not return FFH data, or the last address is accessed.

In the case where the data read is not FFH, another erase operation is performed. (Refer to Set-up Erase/Erase). Verification then resumes from the address of the last-verified byte. Once all bytes in the array have been verified, the erase step is complete. The device can be programmed. At this point, the verify operation is terminated by writing a valid command (e.g. Program Set-up) to the command register. Figure 5, the Quick-Erase algorithm, illustrates how commands and bus operations are combined to perform electrical erasure of the 28F512. Refer to A.C. Erase Characteristics and Waveforms for specific timing parameters.

Set-up Program/Program Commands

Set-up program is a command-only operation that stages the device for byte programming. Writing 40H into the command register performs the set-up operation.

Once the program set-up operation is performed, the next Write-Enable pulse causes a transition to an active programming operation. Addresses are internally latched on the falling edge of the Write-Enable pulse. Data is internally latched on the rising edge of the Write-Enable pulse. The rising edge of Write-Enable also begins the programming operation. The programming operation terminates with the next rising edge of Write-Enable, used to write the program-verify command. Refer to A.C. Program-

ming Characteristics and Waveforms for specific timing parameters.

Program-Verify Command

The 28F512 is programmed on a byte-by-byte basis. Byte programming may occur sequentially or at random. Following each programming operation, the byte just programmed must be verified.

The program-verify operation is initiated by writing C0H into the command register. The register write terminates the programming operation with the rising edge of its Write-Enable pulse. The program-verify operation stages the device for verification of the byte last programmed. No new address information is latched.

The 28F512 applies an internally-generated margin voltage to the byte. A microprocessor read cycle outputs the data. A successful comparison between the programmed byte and true data means that the byte is successfully programmed. Programming then proceeds to the next desired byte location. Figure 4, the 28F512 Quick-Pulse Programming algorithm, illustrates how commands are combined with bus operations to perform byte programming. Refer to A.C. Programming Characteristics and Waveforms for specific timing parameters.

Reset Command

A reset command is provided as a means to safely abort the erase- or program-command sequences. Following either set-up command (erase or program) with two consecutive writes of FFH will safely abort the operation. Memory contents will not be altered. A valid command must then be written to place the device in the desired state.

EXTENDED ERASE/PROGRAM CYCLING

EEPROM cycling failures have always concerned users. The high electrical field required by thin oxide EEPROMs for tunneling can literally tear apart the oxide at defect regions. To combat this, some suppliers have implemented redundancy schemes, reducing cycling failures to insignificant levels. However, redundancy requires that cell size be doubled—an expensive solution.

Intel has designed extended cycling capability into its ETOX-II flash memory technology. Resulting improvements in cycling reliability come without increasing memory cell size or complexity. First, an advanced tunnel oxide increases the charge carrying ability ten-fold. Second, the oxide area per cell subjected to the tunneling electric field is one-tenth that of common EEPROMs, minimizing the probability of oxide defects in the region. Finally, the peak electric field during erasure is approximately 2 MV/cm lower than EEPROM. The lower electric field greatly reduces oxide stress and the probability of failure—increasing time to wearout by a factor of 100,000,000.

The device is programmed and erased using Intel's Quick-Pulse Programming and Quick-Erase algorithms. Intel's algorithmic approach uses a series of operations (pulses), along with byte verification, to completely and reliably erase and program the device.

QUICK-PULSE PROGRAMMING ALGORITHM

The Quick-Pulse Programming algorithm uses programming operations of 10 μs duration. Each operation is followed by a byte verification to determine when the addressed byte has been successfully programmed. The algorithm allows for up to 25 programming operations per byte, although most bytes verify on the first or second operation. The entire sequence of programming and byte verification is performed with V_{PP} at high voltage. Figure 4 illustrates the Quick-Pulse Programming algorithm.

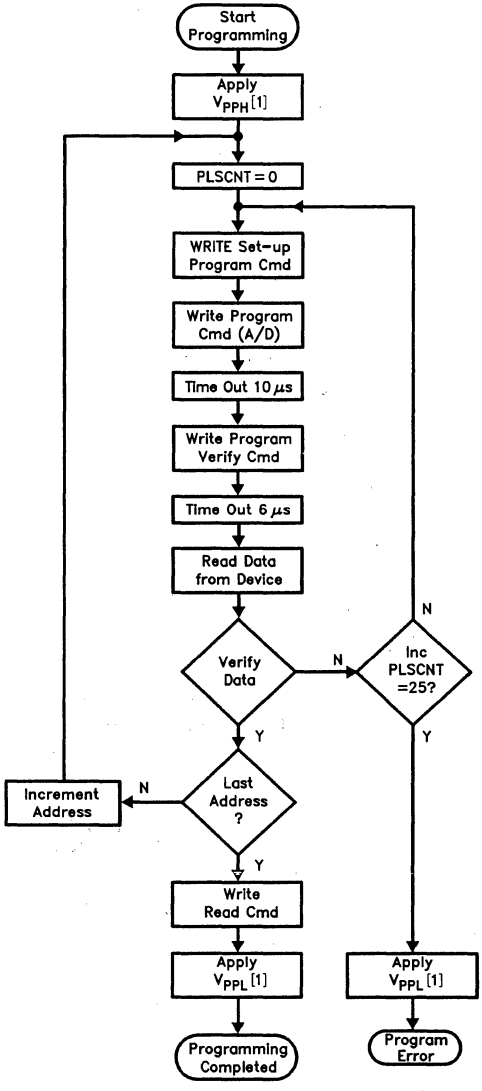
QUICK-ERASE ALGORITHM

Intel's Quick-Erase algorithm yields fast and reliable electrical erasure of memory contents. The algorithm employs a closed-loop flow, similar to the Quick-Pulse Programming algorithm, to simultaneously remove charge from all bits in the array.

Erasure begins with a read of memory contents. The 28F512 is erased when shipped from the factory. Reading FFH data from the device would immediately be followed by device programming.

For devices being erased and reprogrammed, uniform and reliable erasure is ensured by first programming all bits in the device to their charged state (Data = 00H). This is accomplished, using the Quick-Pulse Programming algorithm, in approximately one second.

Erase execution then continues with an initial erase operation. Erase verification (data = FFH) begins at address 0000H and continues through the array to the last address, or until data other than FFH is encountered. With each erase operation, an increasing number of bytes verify to the erased state. Erase efficiency may be improved by storing the address of the last byte verified in a register. Following the next erase operation, verification starts at that stored address location. Erasure typically occurs in one seconds. Figure 5 illustrates the Quick-Erase algorithm.



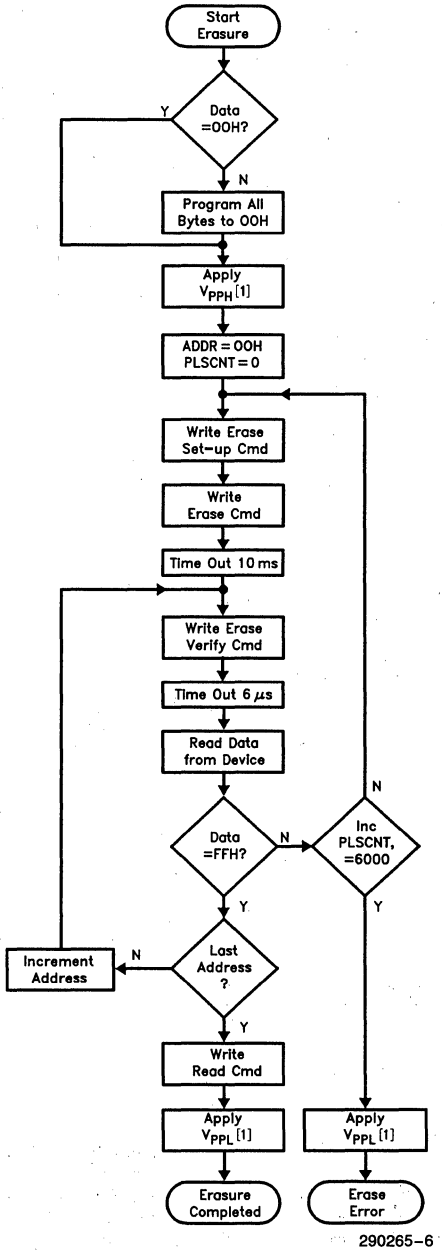
Bus Operation	Command	Comments
Standby		Wait for V_{PP} Ramp to V_{PPH}(1)
		Initialize Pulse-Count
Write	Set-up Program	Data = 40H
Write	Program	Valid Address/Data
Standby		Duration of Program Operation (t_{WHWH1})
Write	Program(3) Verify	Data = C0H; Stops Program Operation(2)
Standby		t_{WHGL}
Read		Read Byte to Verify Programming
Standby		Compare Data Output to Data Expected
Write	Read	Data = 00H, Resets the Register for Read Operations
Standby		Wait for V_{PP} Ramp to V_{PPL}(1)

NOTES:

1. See DC Characteristics for value of V_{PPH}. The V_{PP} power supply can be hard-wired to the device or switchable. When V_{PP} is switched, V_{PPL} may be ground, no-connect with a resistor tied to ground, or less than 6.5V. Refer to Principles of Operation.
2. Refer to Principles of Operation.
3. Program Verify is only performed after byte programming. A final read/compare may be performed (optional) after the register is written with the Read command.
4. **CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.**

Figure 4. 28F512 Quick-Pulse Programming Algorithm

8



Bus Operation	Command	Comments
		Entire memory must = 00H before erasure
		Use Quick-Pulse Programming Algorithm (Figure 4)
Standby		Wait for V_{PP} Ramp to V_{PPH}(1)
		Initialize Addresses and Pulse-Count
Write	Set-up Erase	Data = 20H
Write	Erase	Data = 20H
Standby		Duration of Erase Operation (t_{WHWH2})
Write	Erase(3) Verify	Addr = Byte to Verify; Data = A0H; Stops Erase Operation(2)
Standby		t_{WHGL}
Read		Read Byte to Verify Erasure
Standby		Compare Output to FFH Increment Pulse-Count
Write	Read	Data = 00H, Resets the Register for Read Operations
Standby		Wait for V_{PP} Ramp to V_{PPL}(1)

NOTES:
1. See DC Characteristics for value of V_{PPH}. The V_{PP} power supply can be hard-wired to the device or switchable. When V_{PP} is switched, V_{PPL} may be ground, no-connect with a resistor tied to ground, or less than 6.5V. Refer to Principles of Operation.
2. Refer to Principles of Operation.
3. Erase Verify is performed only after chip-erasure. A final read/compare may be performed (optional) after the register is written with the read command.
4. **CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.**

Figure 5. 28F512 Quick-Erase Algorithm

DESIGN CONSIDERATIONS

Two-Line Output Control

Flash-memories are often used in larger memory arrays. Intel provides two read-control inputs to accommodate multiple memory connections. Two-line control provides for:

a. the lowest possible memory power dissipation and,
b. complete assurance that output bus contention will not occur.

To efficiently use these two control inputs, an address-decoder output should drive chip-enable, while the system's read signal controls all flash-memories and other parallel memories. This assures that only enabled memory devices have active outputs, while deselected devices maintain the low power standby condition.

Power Supply Decoupling

Flash memory power-switching characteristics require careful device decoupling. System designers are interested in three supply current (I_{CC}) issues—standby, active, and transient current peaks produced by falling and rising edges of chip-enable. The capacitive and inductive loads on the device outputs determine the magnitudes of these peaks.

Two-line control and proper decoupling capacitor selection will suppress transient voltage peaks. Each device should have a 0.1 μF ceramic capacitor connected between V_{CC} and V_{SS}, and between V_{PP} and V_{SS}.

Place the high-frequency, low-inherent-inductance capacitors as close as possible to the devices. Also, for every eight devices, a 4.7 μF electrolytic capacitor should be placed at the array's power supply connection, between V_{CC} and V_{SS}. The bulk capacitor will overcome voltage slumps caused by printed-circuit-board trace inductance, and will supply charge to the smaller capacitors as needed.

V_{PP} Trace on Printed Circuit Boards

Programming flash-memories, while they reside in the target system, requires that the printed circuit board designer pay attention to the V_{PP} power supply trace. The V_{PP} pin supplies the memory cell current for programming. Use similar trace widths and layout considerations given the V_{CC} power bus. Adequate V_{PP} supply traces and decoupling will decrease V_{PP} voltage spikes and overshoots.

Power Up/Down Protection

The 28F512 is designed to offer protection against accidental erasure or programming, caused by spurious system-level signals that may exist during power transitions. Also, with its control register architecture, alteration of memory contents only occurs after successful completion of the two-step command sequences. Power supply sequencing is not required. Internal circuitry of the 28F512 ensures that the command register architecture is reset to the read mode on power up.

A system designer must guard against active writes for V_{CC} voltages above the V_{LKO} when V_{PP} is active. Since both WE# and CE# must be low for a command write, driving either to V_{IH} will prohibit writes.

The control register architecture provides an added level of protection since alteration of memory contents only occurs after successful completion of the two-step command sequences.

28F512 Power Dissipation

When designing portable systems, designers must consider battery power consumption not only during device operation, but also for data retention during system idle time. Flash nonvolatility increases the usable battery life of your system because the 28F512 does not consume any power to retain code or data when the system is off. Table 4 illustrates the power dissipated when updating the 28F512.

Table 4. 28F512 Typical Update Power Dissipation(4)

Operation	Power Dissipation (Watt-Seconds)	Notes
Array Program/ Program Verify	0.085	1
Array Erase/ Erase Verify	0.092	2
One Complete Cycle	0.262	3

NOTES:
1. Formula to calculate typical Program/Program Verify Power = [V_{PP} × # Bytes × typical # Prog Pulses (t_{WHWH1} × I_{PP2} typical + t_{WHGL} × I_{PP4} typical)] + [V_{CC} × # Bytes × typical # Prog Pulses (t_{WHWH1} × I_{CC2} typical + t_{WHGL} × I_{CC4} typical)].
2. Formula to calculate typical Erase/Erase Verify Power = [V_{PP} (I_{PP3} typical × t_{ERASE} typical + I_{PP5} typical × t_{WHGL} × # Bytes)] + [V_{CC} (I_{CC3} typical × t_{ERASE} typical + I_{CC5} typical × t_{WHGL} × # Bytes)].
3. Once Complete Cycle = Array Preprogram + Array Erase + Program.
4. "Typicals" are not guaranteed, but based on a limited number of samples from production lots.

ABSOLUTE MAXIMUM RATINGS*

Operating Temperature
During Read ..... −40°C to +125°C(1)
During Erase/Program ..... −40°C to +125°C

Temperature Under Bias ..... −40°C to +125°C

Storage Temperature ..... −65°C to +150°C

Voltage on Any Pin with
Respect to Ground ..... −2.0V to +7.0V(2)

Voltage on Pin A_9 with
Respect to Ground ..... −2.0V to +13.5V(2, 3)

V_{PP} Supply Voltage with
Respect to Ground
During Erase/Program .... −2.0V to +14.0V(2, 3)

V_{CC} Supply Voltage with
Respect to Ground ..... −2.0V to +7.0V(2)

Output Short Circuit Current ..... 100 mA(4)

Maximum Junction Temperature (T_J) ..... 140°C

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for automotive product as defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to −2.0V for periods less than 20 ns. Maximum DC voltage on output pins is V_{CC} + 0.5V, which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.
3. Maximum DC voltage on A_9 or V_{PP} may overshoot to +14.0V for periods less than 20 ns.
4. Output shorted for no more than one second. No more than one output shorted at a time.

OPERATING CONDITIONS

Symbol	Parameter	Limits		Unit	Comments
		Min	Max		
T_A	Operating Temperature	−40	125	°C	For Read-Only and Read/Write Operations
V_{CC}	V_{CC} Supply Voltage	4.50	5.50	V	

DC CHARACTERISTICS—TTL/NMOS COMPATIBLE

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1			±1.0	µA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1			±10	µA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1			1.0	mA	$V_{CC} = V_{CC}$ Max CE# = V_{IH}
I_{CC1}	V_{CC} Active Read Current	1		10	30	mA	$V_{CC} = V_{CC}$ Max, CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Programming Current	1, 2		1.0	30	mA	Programming in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		5.0	30	mA	Erasure in Progress
I_{CC4}	V_{CC} Program Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±10	µA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1		90	200	µA	$V_{PP} > V_{CC}$
					±10	µA	$V_{PP} \leq V_{CC}$

8

DC CHARACTERISTICS—TTL/NMOS COMPATIBLE (Continued)

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typ	Max		
I_{PP2}	V_{PP} Programming Current	1, 2		8.0	30	mA	$V_{PP} = V_{PPH}$ Programming in Progress
I_{PP3}	V_{PP} Erase Current	1, 2		4.0	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Program Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.0		$V_{CC} + 0.5$	V	
V_{OL}	Output Low Voltage				0.45	V	$I_{OL} = 2.1$ mA $V_{CC} = V_{CC}$ Min
V_{OH1}	Output High Voltage		2.4			V	$I_{OH} = -2.5$ mA $V_{CC} = V_{CC}$ Min
V_{ID}	A_9 Intelligent Identifier Voltage		11.50		13.00	V	$A_9 = V_{ID}$
I_{ID}	V_{CC} ID Current	1		10	30	mA	$A_9 = V_{ID}$
	V_{PP} ID Current			90	500	μA	
V_{PPL}	V_{PP} during Read-Only Operations		0.00		6.5	V	**NOTE:** Erase/Program are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} during Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

DC CHARACTERISTICS—CMOS COMPATIBLE

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typ	Max		
I_{LI}	Input Leakage Current	1			±1.0	μA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1			±10	μA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1		50	100	μA	$V_{CC} = V_{CC}$ Max CE# = V_{CC} ±0.2V
I_{CC1}	V_{CC} Active Read Current	1		10	30	mA	$V_{CC} = V_{CC}$ Max, CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Programming Current	1, 2		1.0	30	mA	Programming in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		5.0	30	mA	Erasure in Progress
I_{CC4}	V_{CC} Program Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress

DC CHARACTERISTICS—CMOS COMPATIBLE (Continued)

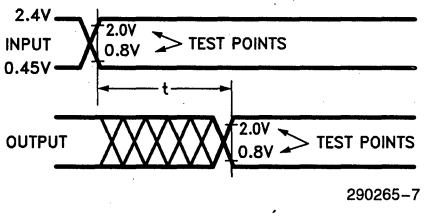
Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typ	Max		
I_{PPS}	V_{PP} Leakage Current	1			±10	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1		90	200	μA	$V_{PP} > V_{CC}$
					±10		$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Programming Current	1, 2		8.0	30	mA	$V_{PP} = V_{PPH}$ Programming in Progress
I_{PP3}	V_{PP} Erase Current	1, 2		4.0	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Program Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{PP5}	V_{PP} Program Erase Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		0.7 V_{CC}		V_{CC} + 0.5	V	
V_{OL}	Output Low Voltage				0.45	V	I_{OL} = 2.1 mA $V_{CC} = V_{CC}$ Min
V_{OH1}	Output High Voltage		0.85 V_{CC}			V	I_{OH} = −2.5 mA, $V_{CC} = V_{CC}$ Min
V_{OH2}			V_{CC} − 0.4				I_{OH} = −100 μA, $V_{CC} = V_{CC}$ Min
V_{ID}	A_9 Intelligent Identifier Voltage		11.50		13.00	V	
I_{ID}	V_{CC} ID Current	1		10	30	mA	A_9 = ID
I_{ID}	V_{PP} ID Current	1		90	500	mA	A_9 = ID
V_{PPL}	V_{PP} during Read-Only Operations		0.00		6.5	V	**NOTE:** Erase/Programs are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} during Read/Write Operations		11.40		12.60	V	V_{PP} = 12.0V
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

CAPACITANCE[3] T_A = 25°C, f = 1.0 MHz

Symbol	Parameter	Limits		Unit	Conditions
		Min	Max		
C_{IN}	Address/Control Capacitance		8	pF	V_{IN} = 0V
C_{OUT}	Output Capacitance		12	pF	V_{OUT} = 0V

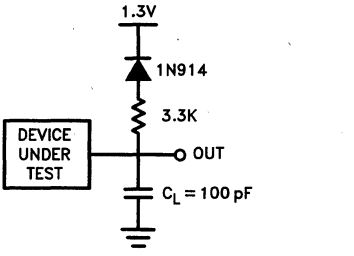
NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. Not 100% tested: characterization data available.
3. Sampled, not 100% tested.
4. "Typicals" are not guaranteed, but are based on a limited number of samples from production lots.

AC TESTING INPUT/OUTPUT WAVEFORM



AC Testing: Inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0". Testing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0". Rise/Fall time ≤ 10 ns.

AC TESTING LOAD CIRCUIT



C_L = 100 pF
C_L includes Jig Capacitance

AC TEST CONDITIONS

Input Rise and Fall Times (10% to 90%) ..... 10 ns
Input Pulse Levels ..... 0.45V and 2.4V
Input Timing Reference Level ..... 0.8V and 2.0V
Output Timing Reference Level ..... 0.8V and 2.0V

AC CHARACTERISTICS—Read-Only Operations(2)

Versions		Notes	28F512-120		28F512-150		Unit
Symbol	Characteristic		Min	Max	Min	Max	
t_{AVAV}/t_{RC}	Read Cycle Time	3	120		150		ns
t_{ELQV}/t_{CE}	Chip Enable Access Time			120		150	ns
t_{AVQV}/t_{ACC}	Address Access Time			120		150	ns
t_{GLQV}/t_{OE}	Output Enable Access Time			50		55	ns
t_{ELQX}/t_{LZ}	Chip Enable to Output in Low Z	3	0		0		ns
t_{EHQZ}	Chip Disable to Output in High Z	3		50		55	ns
t_{GLQX}/t_{OLZ}	Output Enable to Output in Low Z	3	0		0		ns
t_{GHQZ}/t_{DF}	Output Disable to Output in High Z	4		30		35	ns
t_{OH}	Output Hold from Address, CE#, or OE# Change(1)	3	0		0		ns
t_{WHGL}	Write Recovery Time before Read		6		6		μs

NOTES:
1. Whichever occurs first.
2. Rise/Fall Time ≤ 10 ns.
3. Not 100% tested characterization data available.
4. Guaranteed by design.

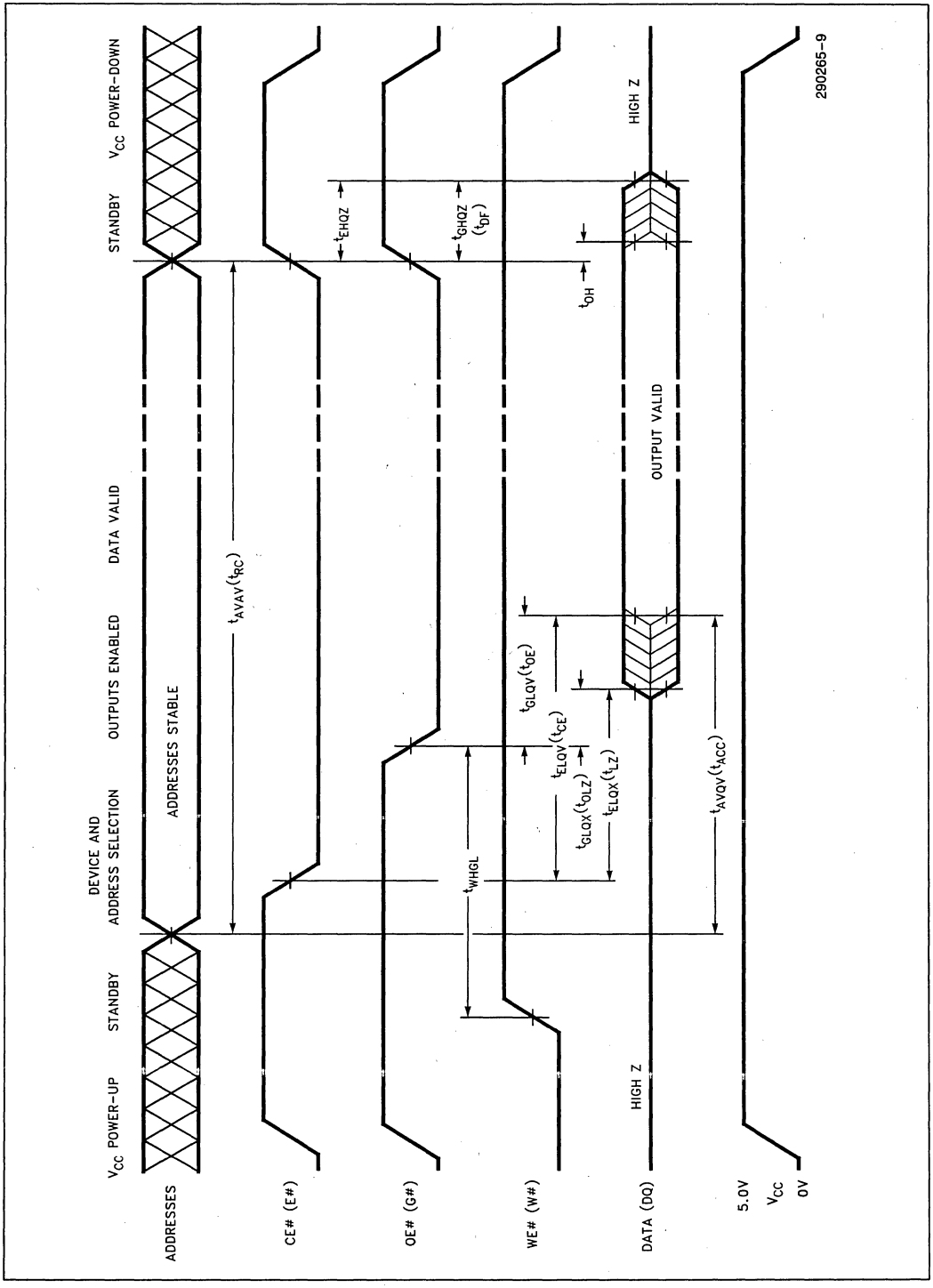


Figure 6. AC Waveforms for Read Operations

AC CHARACTERISTICS—Write/Erase/Program Operations[1, 3]

Versions		Notes	28F512-120		28F512-150		Unit
Symbol	Characteristic		Min	Max	Min	Max	
t_{AVAV}/t_{WC}	Write Cycle Time		120		150		ns
t_{AVWL}/t_{AS}	Address Set-Up Time		0		0		ns
t_{WLAX}/t_{AH}	Address Hold Time	2	60		60		ns
t_{DVWH}/t_{DS}	Data Set-up Time		50		50		ns
t_{WHDX}/t_{DH}	Data Hold Time		10		10		ns
t_{WHGL}	Write Recovery Time before Read		6		6		μs
t_{GHWL}	Read Recovery Time before Write		0		0		μs
t_{ELWL}/t_{CS}	Chip Enable Set-Up Time before Write	2	20		20		ns
t_{WHEH}/t_{CH}	Chip Enable Hold Time		0		0		ns
t_{WLWH}/t_{WP}	Write Pulse Width[2]	2	80		80		ns
t_{ELEH}	Alternative Write[2] Pulse Width	2	80		80		ns
t_{WHWL}/t_{WPH}	Write Pulse Width High		20		20		ns
t_{WHWH1}	Duration of Programming Operation	4	10		10		μs
t_{WHWH2}	Duration of Erase Operation	4	9.5		9.5		ms
t_{VPEL}	V_{PP} Set-Up Time to Chip Enable Low		1.0		1.0		ms

NOTES:
1. Read timing characteristics during read/write operations are the same as during read-only operations. Refer to AC Characteristics for Read-Only Operations.
2. Chip-Enable Controlled Writes: Write operations are driven by the valid combination of Chip-Enable and Write-Enable. In systems where Chip-Enable defines the write pulse width (within a longer Write-Enable timing waveform) all set-up, hold, and inactive Write-Enable times should be measured relative to the Chip-Enable waveform.
3. Rise/Fall time ≤ 10 ns.
4. The integrated stop timer terminates the programming/erase operations, thereby eliminating the need for a maximum specification.

ERASE AND PROGRAMMING PERFORMANCE

Parameter	Limits			Notes	Unit	Comments
	Min	Typ	Max			
Chip Erase Time		1	60	1, 3, 4, 6	Sec	Excludes 00H Programming Prior to Erasure
Chip Program Time		1	6.25	1, 2, 4	Sec	Excludes System-Level Overhead
Erase/Program Cycles	1,000	100,000		1, 5	Cycles	

NOTES:
1. "Typicals" are not guaranteed, but based on a limited number of samples taken from production lots. Data taken at $T = 25°C$, $V_{PP} = 12.0V$, $V_{CC} = 5.0V$.
2. Minimum byte programming time excluding system overhead is 16 μs (10 μs program + 6 μs write recovery), while maximum is 400 μs/byte (16 μs × 25 loops allowed by algorithm). Max chip programming time is specified lower than the worst case allowed by the programming algorithm since most bytes program significantly faster than the worst case byte.
3. Excludes 00H programming prior to erasure.
4. Excludes system-level overhead.
5. Refer to RR-60 "ETOX Flash Memory Reliability Data Summary" for typical cycling data and failure rate calculations.
6. Maximum erase specification is determined by algorithmic limit and accounts for cumulative effect of erasure at $T = -40°C$, 1,000 cycles, $V_{PP} = 11.4V$, $V_{CC} = 4.5V$.

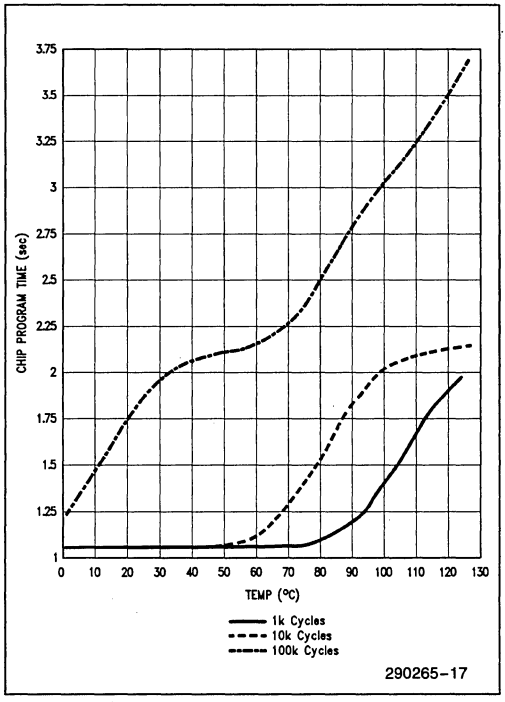


Figure 7. 28F512 Typical Program Time at 12V

See Note 1, Page 19.

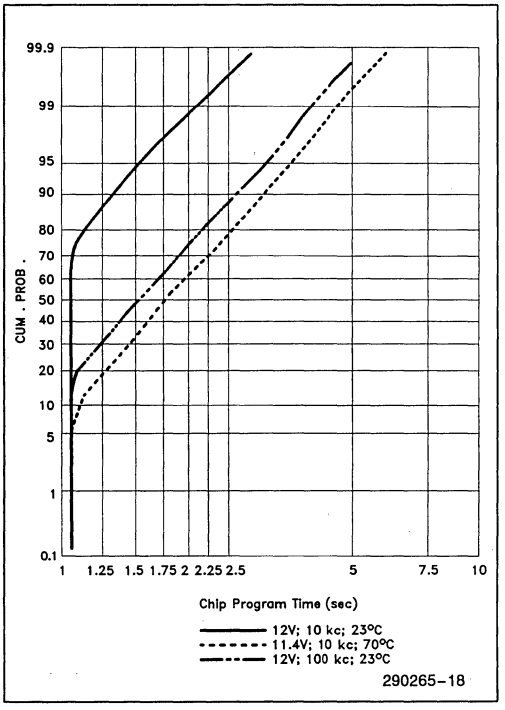
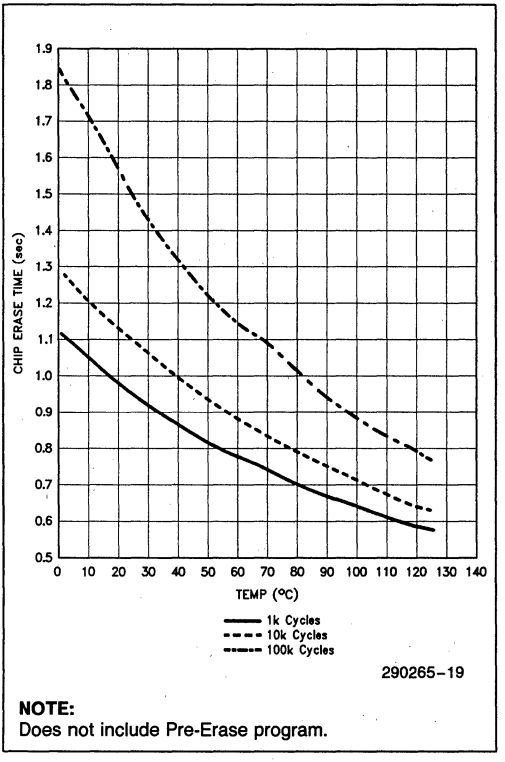


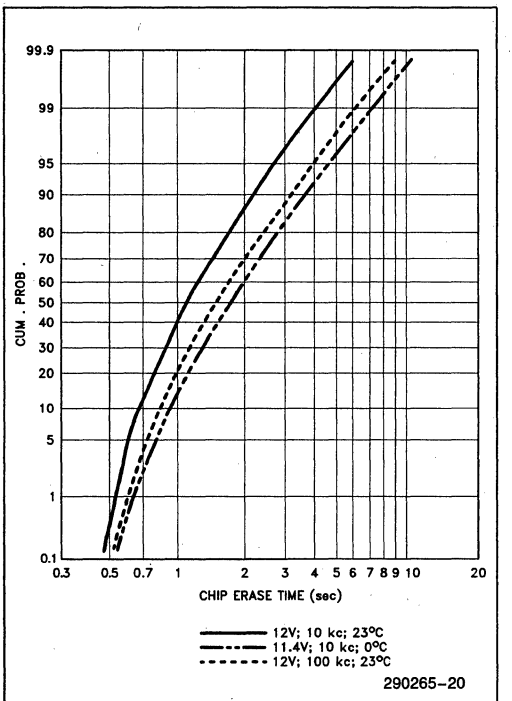
Figure 8. 28F512 Typical Programming Capability

8



NOTE:
Does not include Pre-Erase program.

Figure 9. 28F512 Typical Erase Time at 12V
See Note 1, Page 19.



NOTE:
Does not include Pre-Erase program.

Figure 10. 28F512 Typical Erase Capability

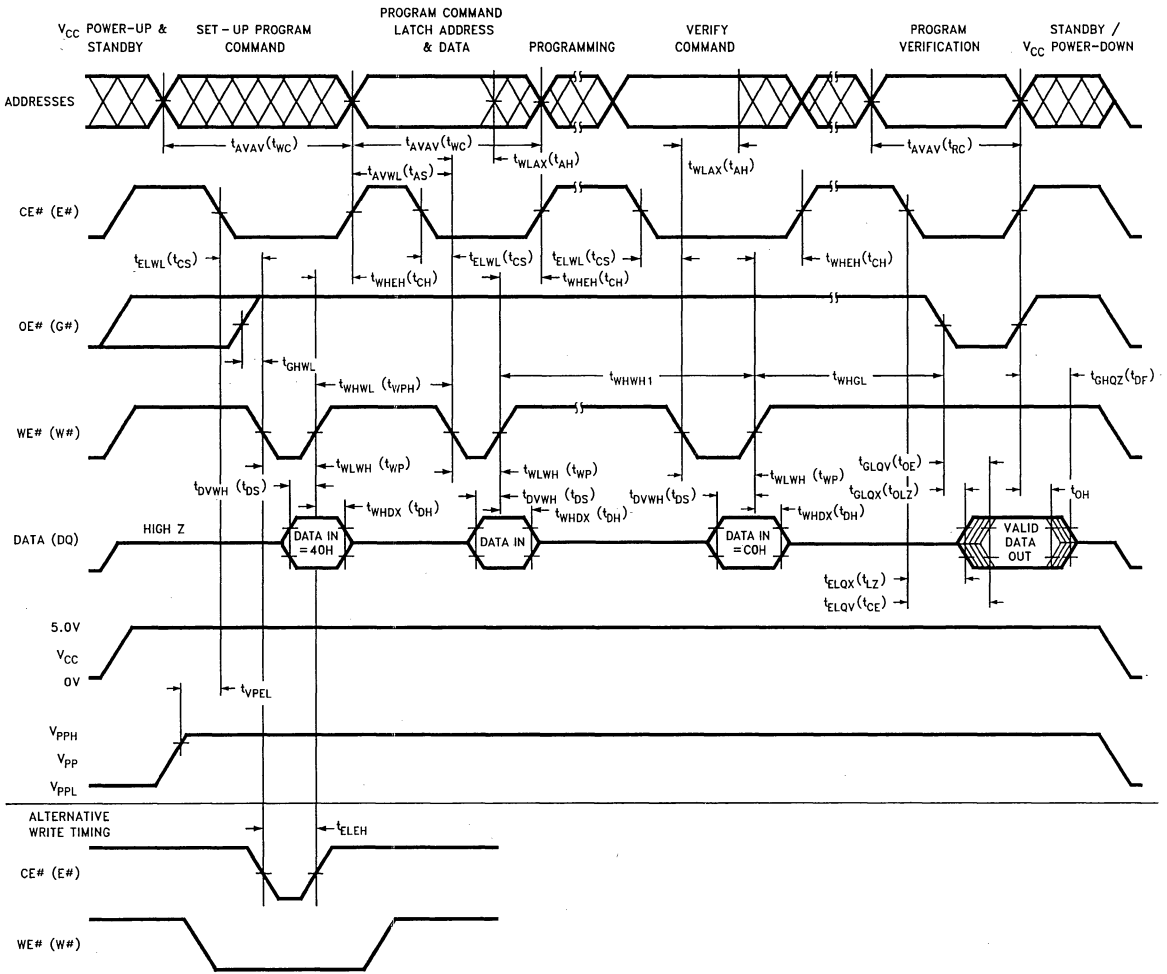


Figure 11. AC Waveforms for Programming Operations

8

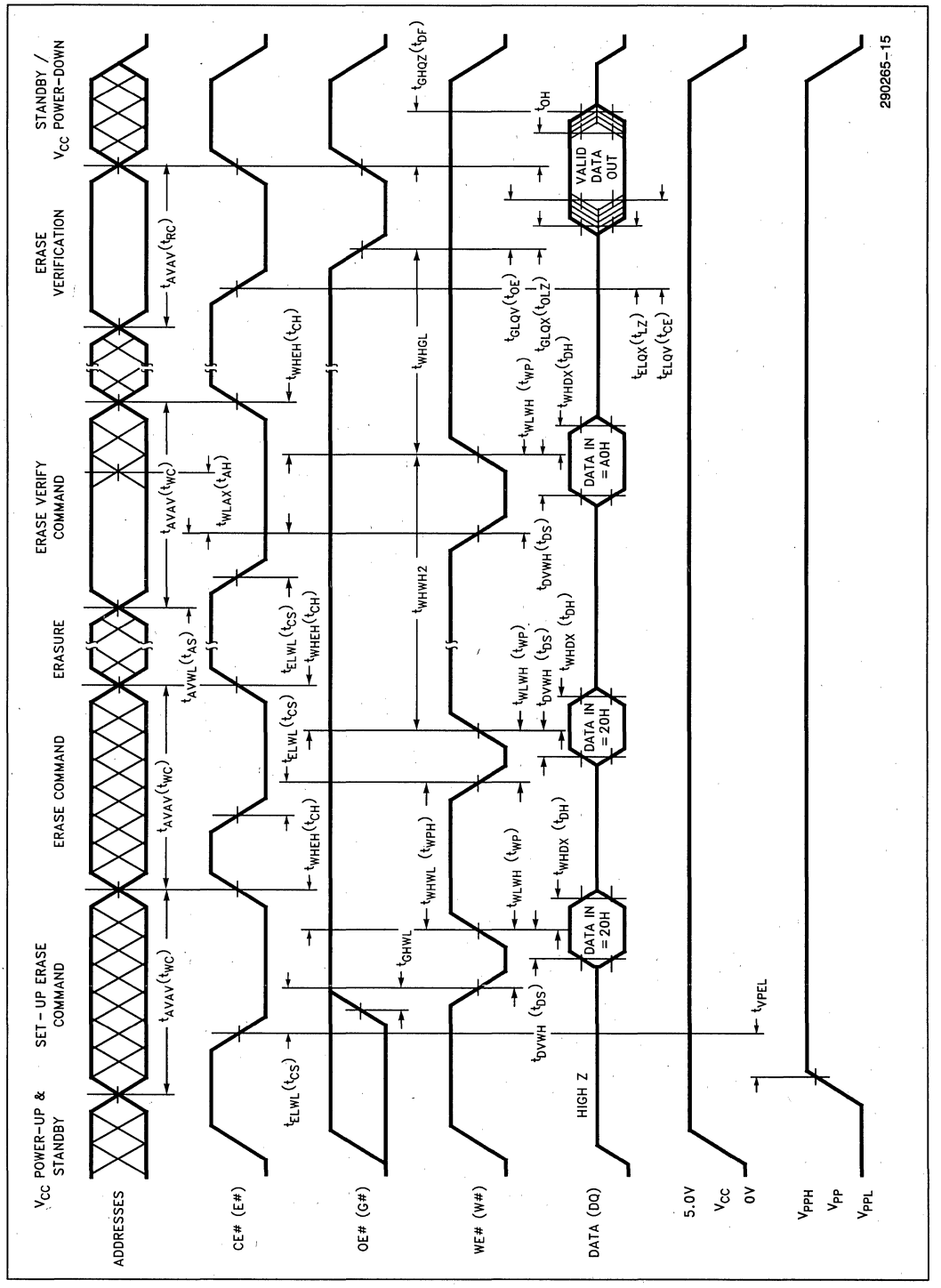
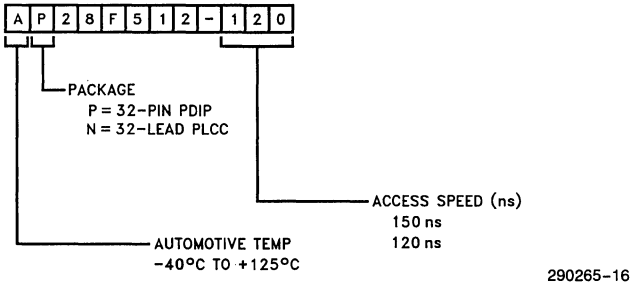


Figure 12. AC Waveforms for Erase Operations

Ordering Information



Valid Combinations:

AP28F512-150	AN28F512-150
AP28F512-120	AN28F512-120

ADDITIONAL INFORMATION

	Order Number
ER-20, "ETOX™ II Flash Memory Technology"	294005
ER-23, "The Intel A28F512 Flash Memory"	294007
RR-60, "ETOX™ Flash Memory Reliability Data Summary"	293002
AP-316, "Using Flash Memory for In-System Reprogrammable Nonvolatile Storage"	292046
AP-325 "Guide to Flash Memory Reprogramming"	292059

REVISION HISTORY

Number	Description
002	Changed Erase/Program Cycles to 1,000 minimum.
003	Added 120 ns speed

8

A28F256A
256K (32K x 8) CMOS FLASH MEMORY

Automotive

- **Extended Automotive Temperature Range −40°C to +125°C**
- **Flash Electrical Chip-Erase**
  - **1 Second Typical Chip-Erase**
- **Quick-Pulse Programming Algorithm**
  - **10 μs Typical Byte-Program**
  - **0.5 Second Chip-Program**
- **1,000 Erase/Program Cycles Minimum Over Automotive Temperature Range**
- **12.0V ±5% V_{PP}**
- **High-Performance Read**
  - **120/150 ns Maximum Access Time**
- **CMOS Low Power Consumption**
  - **30 mA Maximum Active Current**
  - **100 μA Maximum Standby Current**
- **Integrated Program/Erase Stop Timer**
- **Command Register Architecture for Microprocessor/Microcontroller Compatible Write Interface**
- **Noise Immunity Features**
  - **±10% V_{CC} Tolerance**
  - **Maximum Latch-Up Immunity through EPI Processing**
- **ETOX™ II Flash Nonvolatile Memory Technology**
  - **EPROM-Compatible Process Base**
  - **High-Volume Manufacturing Experience**
- **JEDEC-Standard Pinouts**
  - **32-Pin Plastic-DIP**
  - **32-Lead PLCC**

  (See Packaging Spec., Order #231369)

Intel's 28F256A CMOS flash memory offers the most cost-effective and reliable alternative for updatable non-volatile memory. The 28F256A adds electrical chip-erasure and reprogramming to familiar EPROM technology. Memory contents can be erased and reprogrammed: in a test socket; in a PROM-programmer socket; on-board during subassembly test; in-system during final test; and in-system after-sale. The 28F256A increases memory flexibility, while contributing to time- and cost-savings. The 28F256A is targeted for alterable code- or data-storage applications where traditional EEPROM functionality (byte-erasure) is either not required or not cost-effective. The 28F256A can also be applied where EPROM ultraviolet erasure is impractical or time-consuming.

The 28F256A is a 256-kilobit non-volatile memory organized as 32768 bytes of 8 bits. Intel's 28F256A is offered in 32-pin Plastic-DIP or 32-lead PLCC packages. Pin assignments conform to JEDEC standards.

Extended erase and program cycling capability is designed into Intel's ETOX™ II (EPROM Tunnel Oxide) process technology. Advanced oxide processing, an optimized tunneling structure, and lower electric field combine to extend reliable cycling beyond that of traditional EEPROMs. With the 12.0V V_{PP} supply, the 28F256A performs a minimum of 1,000 erase and program cycles well within the time limits of the Quick-Pulse Programming and Quick-Erase algorithms.

Intel's 28F256A employs advanced CMOS circuitry for systems requiring high-performance access speeds, low power consumption, and immunity to noise. Its 150 nanosecond access time provides no-WAIT-state performance for a wide range of microprocessors and microcontrollers. Maximum standby current of 100 μA translates into power savings when the device is deselected. Finally, the highest degree of latch-up protection is achieved through Intel's unique EPI processing. Prevention of latch-up is provided for stresses up to 100 mA on address and data pins, from −1V to V_{CC} + 1V.

With Intel's ETOX-II process base, the 28F256A leverages years of EPROM experience to yield the highest levels of quality, reliability, and cost-effectiveness.

In order to meet the rigorous environmental requirements of Automotive Applications, Intel offers the 28F256A in extended Automotive temperature range. Read and Write Characteristics are guaranteed over the range of −40°C to +125°C ambient.

March 1994
Order Number: 290168-006

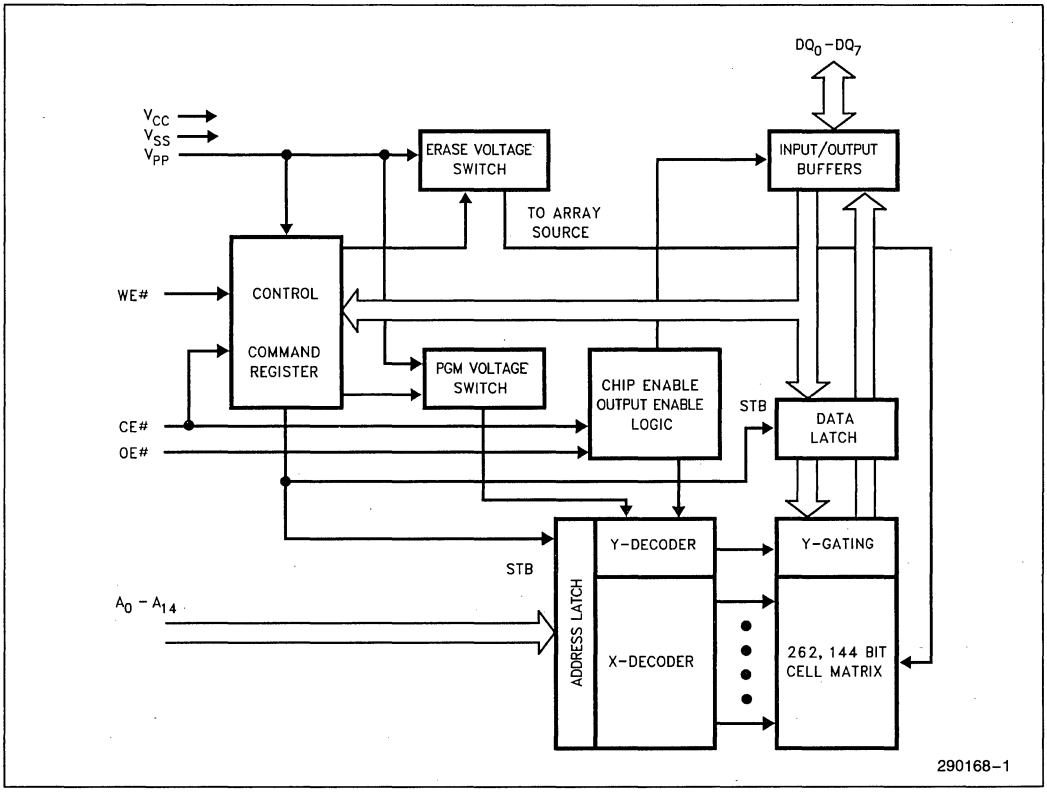


Figure 1. 28F256A Block Diagram

AUTOMOTIVE TEMPERATURE FLASH MEMORIES

The Intel Automotive Flash Memories have received additional processing to enhance product characteristics. The Automotive temperature range is −40°C to +125°C during the read/write/erase/program operations.

Speed Versions	Packaging Options	
	Plastic-DIP	PLCC
-120	AP	AN
-150	AP	AN

8

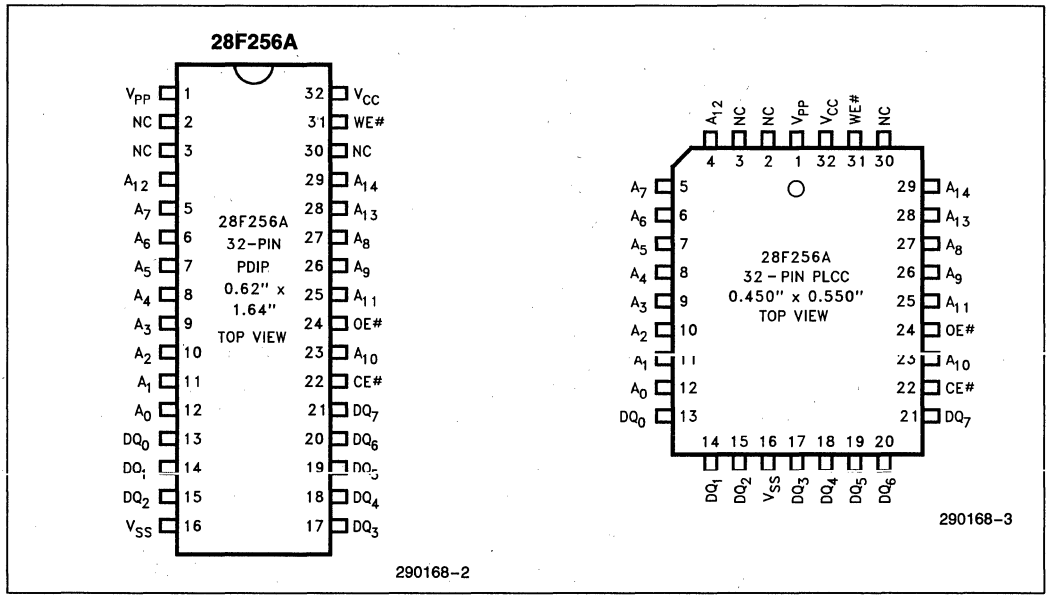


Figure 2. 28F256A Pin Configurations

Table 1. Pin Description

Symbol	Type	Name and Function
A_0–A_{14}	INPUT	**ADDRESS INPUTS** for memory addresses. Addresses are internally latched during a write cycle.
DQ_0–DQ_7	INPUT/OUTPUT	**DATA INPUT/OUTPUT:** Inputs data during memory write cycles; outputs data during memory read cycles. The data pins are active high and float to tri-state OFF when the chip is deselected or the outputs are disabled. Data is internally latched during a write cycle.
CE#	INPUT	**CHIP ENABLE:** Activates the device's control logic, input buffers, decoders and sense amplifiers. CE# is active low; CE high deselects the memory device and reduces power consumption to standby levels.
OE#	INPUT	**OUTPUT ENABLE:** Gates the devices output through the data buffers during a read cycle. OE# is active low.
WE#	INPUT	**WRITE ENABLE:** Controls writes to the control register and the array. Write enable is active low. Addresses are latched on the falling edge and data is latched on the rising edge of the WE# pulse. **Note:** With $V_{PP} \leq 6.5V$, memory contents cannot be altered.
V_{PP}		**ERASE/PROGRAM POWER SUPPLY** for writing the command register, erasing the entire array, or programming bytes in the array.
V_{CC}		**DEVICE POWER SUPPLY** (5V ±10%)
V_{SS}		**GROUND**
NC		**NO INTERNAL CONNECTION** to device. Pin may be driven or left floating.

APPLICATIONS

The 28F256A flash memory adds electrical chip-erasure and reprogrammability to EPROM non-volatility and ease of use. The 28F256A is ideal for storing code or data-tables in applications where periodic updates are required. The 28F256A also serves as a dense, nonvolatile data acquisition and storage medium.

The need for code updates pervades all phases of a system's life—from prototyping to system manufacture to after-sale service. In the factory, during prototyping, revisions to control code necessitate ultraviolet erasure and reprogramming of EPROM-based prototype codes. The 28F256A replaces the 15- to 20-minute ultraviolet erasure with one-second electrical erasure. Electrical chip-erasure and reprogramming occur in the same workstation or PROM-programmer socket.

Diagnostics, performed at subassembly or final assembly stages, often require the socketing of EPROMs. Socketed test codes are ultimately replaced with EPROMs containing the final program. With electrical chip-erasure and reprogramming, the 28F256A is soldered to the circuit board. Test codes are programmed into the 28F256A as it resides on the circuit board. Ultimately, the final code can be downloaded to the device. The 28F256A's in-circuit alterability eliminates unnecessary handling and less-reliable socketed connections, while adding greater test flexibility.

Material and labor costs associated with code changes increase at higher levels of system integration—the most costly being code updates after sale. Code "bugs", or the desire to augment system functionality, prompt after-sale code updates. Field revisions to EPROM-based code require the removal of EPROM components or entire boards.

Designing with the in-circuit alterable 28F256A eliminates socketed memories, reduces overall material costs, and drastically cuts the labor costs associated with code updates. With the 28F256A, code updates are implemented locally via an edge-connector, or remotely over a serial communication link.

The 28F256A's electrical chip-erasure, byte reprogrammability, and complete nonvolatility fit well with data accumulation needs. Electrical chip-erasure gives the designer a "blank-slate" in which to log data. Data can be periodically off-loaded for analysis—erasing the slate and repeating the cycle. Or, multiple devices can maintain a "rolling window" of accumulated data.

A high degree of on-chip feature integration simplifies memory-to-processor interfacing. Figure 3 illustrates the interface between the MCS®-51 microcontroller and one 28F256A flash memory in a minimum chip-count system. Figure 4 depicts two 28F256As tied to the 80C186 system bus. In both instances, the 28F256A's architecture minimizes interface circuitry needed for complete in-circuit updates of memory contents. (Comprehensive system design information is included in AP-316, "Using

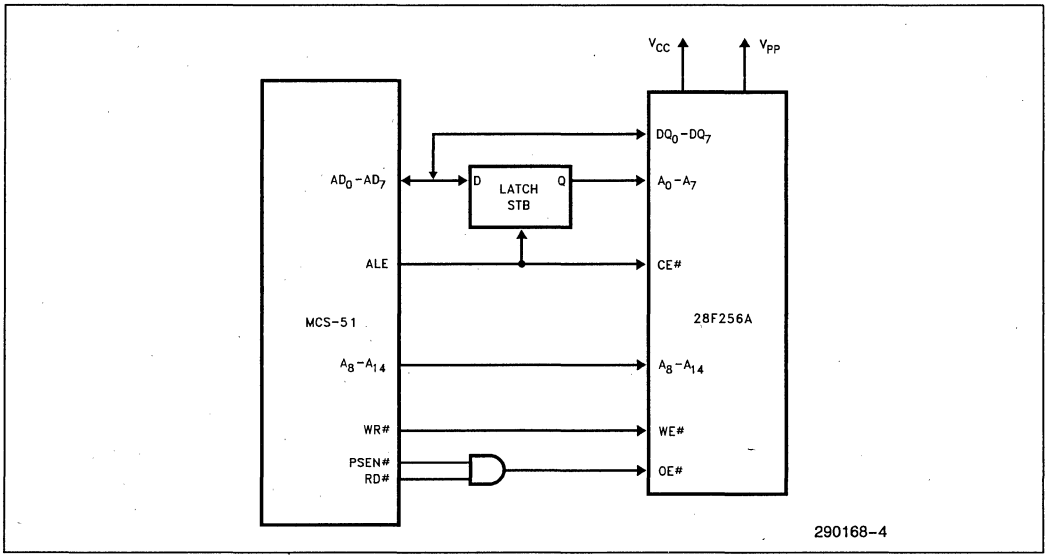


Figure 3. 28F256A in an MCS®-51 System

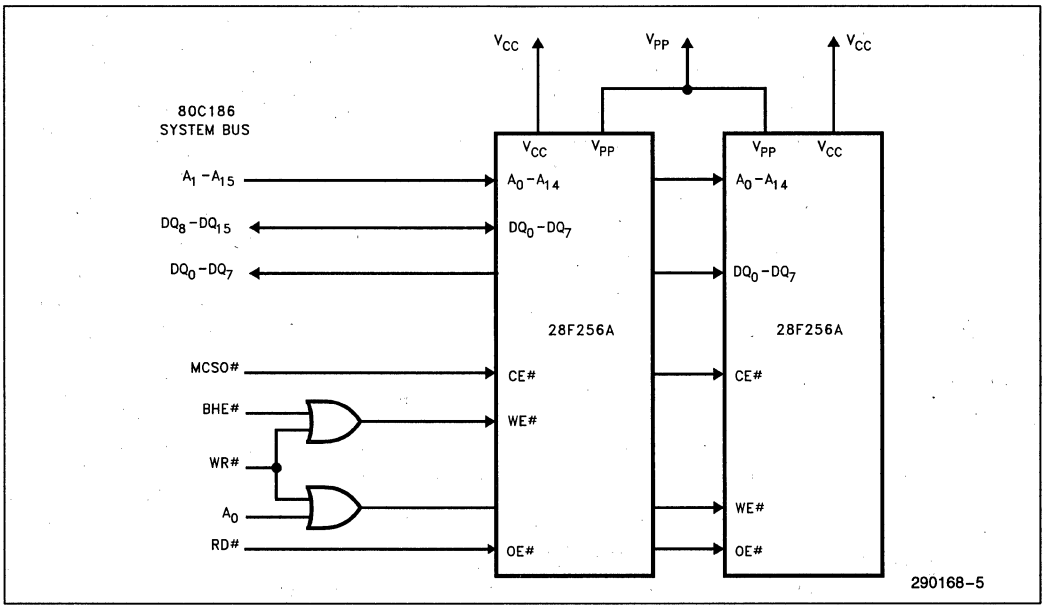


Figure 4. 28F256A in an 80C186 System

the 28F256A Flash Memory for In-System Reprogrammable Nonvolatile Storage", Order Number 292046-002).

With cost-effective in-system reprogramming and extended cycling capability, the 28F256A fills the functionality gap between traditional EPROMs and E²PROMs. EPROM-compatible specifications, straightforward interfacing, and in-circuit alterability allows designers to easily augment memory flexibility and satisfy the need for updatable nonvolatile storage in today's designs.

PRINCIPLES OF OPERATION

Flash-memory augments EPROM functionality with in-circuit electrical erasure and reprogramming. The 28F256A introduces a command register to manage this new functionality. The command register allows for: 100% TTL-level control inputs; fixed power supplies during erasure and programming; and maximum EPROM compatibility.

In the absence of high voltage on the V_{PP} pin, the 28F256A is a read-only memory. Manipulation of the external memory-control pins yields the standard EPROM read, standby, output disable, and intelligent identifier operations.

The same EPROM read, standby, and output disable operations are available when high voltage is applied to the V_{PP} pin. In addition, high voltage on V_{PP} enables erasure and programming of the device. All functions associated with altering memory contents—intelligent identifier, erase, erase verify, program, and program verify—are accessed via the command register.

Commands are written to the register using standard microprocessor write timings. Register contents serve as input to an internal state-machine which controls the erase and programming circuitry. Write cycles also internally latch addresses and data needed for programming or erase operations. With the appropriate command written to the register, standard microprocessor read timings output array data, access the intelligent identifier codes, or output data for erase and program verification.

Integrated Program/Erase Stop Timer

Successive command write cycles define the durations of program and erase operations; specifically, the program or erase time durations are normally terminated by associated program or erase verify commands. An integrated stop timer provides simplified timing control over these operations; thus eliminating the need for maximum program/erase timing specifications. Programming and erase pulse durations are minimums only. When the stop timer terminates a program or erase operation, the device enters an inactive state and remains inactive until receiving the appropriate verify or reset command.

Table 2. 28F256A Bus Operations

Operation	Pins	V_{PP}(1)	A_0	A_9	CE#	OE#	WE#	DQ_0–DQ_7
READ-ONLY	Read	V_{PPL}	A_0	A_9	V_{IL}	V_{IL}	V_{IH}	Data Out
	Output Disable	V_{PPL}	X	X	V_{IL}	V_{IH}	V_{IH}	Tri-State
	Standby	V_{PPL}	X	X	V_{IH}	X	X	Tri-State
	intelligent ID Manufacturer(2)	V_{PPL}	V_{IL}	V_{ID}(3)	V_{IL}	V_{IL}	V_{IH}	Data = 89H
	intelligent ID Device(2)	V_{PPL}	V_{IH}	V_{ID}(3)	V_{IL}	V_{IL}	V_{IH}	Data = B9H
READ/WRITE	Read	V_{PPH}	A_0	A_9	V_{IL}	V_{IL}	V_{IH}	Data Out(4)
	Output Disable	V_{PPH}	X	X	V_{IL}	V_{IH}	V_{IH}	Tri-State
	Standby(5)	V_{PPH}	X	X	V_{IH}	X	X	Tri-State
	Write	V_{PPH}	A_0	A_9	V_{IL}	V_{IH}	V_{IL}	Data In(6)

NOTES:
1. V_{PPL} may be ground, a no-connect with a resistor tied to ground, or ≤ 6.5V. V_{PPH} is the programming voltage specified for the device. Refer to D.C. Characteristics. When $V_{PP} = V_{PPL}$ memory contents can be read but not written or erased.
2. Manufacturer and device codes may also be accessed via a command register write sequence. Refer to Table 3. All other addresses low.
3. V_{ID} is the intelligent identifier high voltage. Refer to DC Characteristics.
4. Read operations with $V_{PP} = V_{PPH}$ may access array data or the intelligent ID.
5. With V_{PP} at high voltage, the standby current equals $I_{CC} + I_{PP}$ (standby).
6. Refer to Table 3 for valid Data-In during a write operation.
7. X can be V_{IL} or V_{IH}.

Write Protection

The command register is only alterable when V_{PP} is at high voltage. Depending upon the application, the system designer may choose to make the V_{PP} power supply switchable—available only when memory updates are desired. When high voltage is removed, the contents of the register default to the read command, making the 28F256A a read-only memory. Memory contents cannot be altered.

Or, the system designer may choose to "hardwire" V_{PP}, making the high voltage supply constantly available. In this instance, all operations are performed in conjunction with the command register. The 28F256A is designed to accommodate either design practice, and to encourage optimization of the processor-memory interface.

The two-step Program/Erase write sequence to the command register provides additional software write protection.

BUS OPERATIONS

Read

The 28F256A has two control functions, both of which must be logically active, to obtain data at the outputs. Chip-Enable (CE#) is the power control and should be used for device selection. Output-Enable (OE#) is the output control and should be used to gate data from the output pins, independent of device selection. Figure 7 illustrates read timing waveforms.

When V_{PP} is low (V_{PPL}), the read only operation is active. This permits reading the data in the array and outputting the intelligent identifier codes (See Table 2). When V_{PP} is high (V_{PPH}), the default condition of the device is the read-only mode. This allows reading the data in the array. Further functionality is achieved through the Command Register as shown in Table 3.

Output Disable

With Output-Enable at a logic-high level (V_{IH}), output from the device is disabled. Output pins are placed in a high-impedance state.

Standby

With Chip-Enable at a logic-high level, the standby operation disables most of the 28F256A's circuitry and substantially reduces device power consumption. The outputs are placed in a high-impedance state, independent of the Output-Enable signal. If the 28F256A is deselected during erasure, programming, or program/erase verification, the device draws active current until the operation is terminated.

Intelligent Identifier Operation

The Intelligent Identifier operation outputs the manufacturer code (89H) and device code (B9H). Programming equipment automatically matches the device with its proper erase and programming algorithms.

With Chip-Enable and Output-Enable at a logic low level, raising A9 to high voltage, V_{ID} (See DC Characteristics), activates the operation. Data read from locations 0000H and 0001H represent the manufacturer's code and the device code, respectively.

The manufacturer- and device-codes can also be read via the command register, for instances where the 28F256A is erased and reprogrammed in the target system. Following a write of 90H to the command register, a read from address location 0000H outputs the manufacturer code (89H). A read from address 0001H outputs the device code (B9H).

Write

Device erasure and programming are accomplished via the command register, when high voltage is applied to the V_{PP} pin. The contents of the register serve as input to the internal state-machine. The state-machine outputs dictate the function of the device.

The command register itself does not occupy an addressable memory location. The register is a latch used to store the command, along with address and data information needed to execute the command. The command register is written by bringing Write-Enable to a logic-low level (V_{IL}), while Chip-Enable is low. Addresses are latched on the falling edge of Write-Enable, while data is latched on the rising edge of the Write-Enable pulse. Standard microprocessor write timings are used.

Refer to A.C. Write Characteristics and the Erase/Programming Waveforms for specific timing parameters.

COMMAND DEFINITIONS

When low voltage is applied to the V_{PP} pin, the contents of the command register default to 00H, enabling read-only operations.

Placing high voltage on the V_{PP} pin enables read/write operations. Device operations are selected by writing specific data patterns into the command register. Table 3 defines these 28F256A register commands.

Table 3. Command Definitions

Command	Bus Cycles Req'd	First Bus Cycle			Second Bus Cycle		
		Operation(1)	Address(2)	Data(3)	Operation(1)	Address(2)	Data(3)
Read Memory	1	Write	X	00H			
Read Intelligent ID(4)	1	Write	X	90H	Read	IA	ID
Set-up Erase/Erase(5)	2	Write	X	20H	Write	X	20H
Erase Verify(5)	2	Write	EA	A0H	Read	X	EVD
Set-up Program/Program(6)	2	Write	X	40H	Write	PA	PD
Program Verify(6)	2	Write	X	C0H	Read	X	PVD
Reset(7)	2	Write	X	FFH	Write	X	FFH

NOTES:
1. Bus operations are defined in Table 2.
2. IA = Identifier address: 00H for manufacturer code, 01H for device code.
   EA = Address of memory location to be read during erase verify.
   PA = Address of memory location to be programmed.
   Addresses are latched on the falling edge of the Write-Enable pulse.
3. ID = Data read from location IA during device identification (Mfr = 89H, Device = B9H).
   EVD = Data read from location EA during erase verify.
   PD = Data to be programmed at location PA. Data is latched on the rising edge of Write-Enable.
   PVD = Data read from location PA during program verify. PA is latched on the Program command.
4. Following the Read Intelligent ID command, two read operations access manufacturer and device codes.
5. Figure 6 illustrates the Quick-Erase Algorithm.
6. Figure 5 illustrates the Quick-Pulse Programming Algorithm.
7. The second bus cycle must be followed by the desired command register write.

Read Command

While V_{PP} is high, for erasure and programming, memory contents can be accessed via the read command. The read operation is initiated by writing 00H into the command register. Microprocessor read cycles retrieve array data. The device remains enabled for reads until the command register contents are altered.

The default contents of the register upon V_{PP} power-up is 00H. This default value ensures that no spurious alteration of memory contents occurs during the V_{PP} power transition. Where the V_{PP} supply is hard-wired to the 28F256A, the device powers-up and remains enabled for reads until the command-register contents are changed. Refer to the A.C. Read Characteristics and Waveforms for specific timing parameters.

Intelligent Identifier Command

Flash memories are intended for use in applications where the local CPU alters memory contents. As such, manufacturer- and device-codes must be accessible while the device resides in the target system. PROM programmers typically access signature codes by raising A9 to a high voltage. However, multiplexing high voltage onto address lines is not a desired system-design practice.

The 28F256A contains an Intelligent Identifier operation to supplement traditional PROM-programming methodology. The operation is initiated by writing 90H into the command register. Following the command write, a read cycle from address 0000H retrieves the manufacturer code of 89H. A read cycle from address 0001H returns the device code of B9H. To terminate the operation, it is necessary to write another valid command into the register.

Set-up Erase/Erase Commands

Set-up Erase is a command-only operation that stages the device for electrical erasure of all bytes in the array. The set-up erase operation is performed by writing 20H to the command register.

To commence chip-erasure, the erase command (20H) must again be written to the register. The erase operation begins with the rising edge of the Write-Enable pulse and terminates with the rising edge of the next Write-Enable pulse (i.e., Erase-Verify Command).

This two-step sequence of set-up followed by execution ensures that memory contents are not accidentally erased. Also, chip-erasure can only occur when high voltage is applied to the V_{PP} pin. In the absence of this high voltage, memory contents are protected against erasure. Refer to A.C. Erase Characteristics and Waveforms for specific timing parameters.

Erase-Verify Command

The erase command erases all bytes of the array in parallel. After each erase operation, all bytes must be verified. The erase verify operation is initiated by writing A0H into the command register. The address for the byte to be verified must be supplied as it is latched on the falling edge of the Write-Enable pulse. The register write terminates the erase operation with the rising edge of its Write-Enable pulse.

The 28F256A applies an internally-generated margin voltage to the addressed byte. Reading FFH from the addressed byte indicates that all bits in the byte are erased.

The erase-verify command must be written to the command register prior to each byte verification to latch its address. The process continues for each byte in the array until a byte does not return FFH data, or the last address is accessed.

In the case where the data read is not FFH, another erase operation is performed. (Refer to Set-up Erase/Erase). Verification then resumes from the address of the last-verified byte. Once all bytes in the array have been verified, the erase step is complete. The device can be programmed. At this point, the verify operation is terminated by writing a valid command (e.g. Program Set-up) to the command register. Figure 6, the Quick-Erase Algorithm, illustrates how commands and bus operations are combined to perform electrical erasure of the 28F256A. Refer to A.C. Erase Characteristics and Waveforms for specific timing parameters.

Set-up Program/Program Commands

Set-up program is a command-only operation that stages the device for byte programming. Writing 40H into the command register performs the set-up operation.

Once the program set-up operation is performed, the next Write-Enable pulse causes a transition to an active programming operation. Addresses are internally latched on the falling edge of the Write-Enable pulse. Data is internally latched on the rising edge of the Write-Enable pulse. The rising edge of Write-Enable also begins the programming operation. The programming operation terminates with the next rising edge of Write-Enable, used to write the program-verify command. Refer to A.C. Programming Characteristics and Waveforms for specific timing parameters.

Program-Verify Command

The 28F256A is programmed on a byte-by-byte basis. Byte programming may occur sequentially or at random. Following each programming operation, the byte just programmed must be verified.

The program-verify operation is initiated by writing C0H into the command register. The register write terminates the programming operation with the rising edge of its Write-Enable pulse. The program-verify operation stages the device for verification of the byte last programmed. No new address information is latched.

The 28F256A applies an internally-generated margin voltage to the byte. A microprocessor read cycle outputs the data. A successful comparison between the programmed byte and true data means that the byte is successfully programmed. Programming then proceeds to the next desired byte location. Figure 5, the 28F256A Quick-Pulse Programming Algorithm, illustrates how commands are combined with bus operations to perform byte programming. Refer to A.C. Programming Characteristics and Waveforms for specific timing parameters.

Reset Command

A reset command is provided as a means to safely abort the erase- or program-command sequences. Following either set-up command (erase or program) with two consecutive writes of FFH will safely abort the operation. Memory contents will not be altered. A valid command must then be written to place the device in the desired state.

EXTENDED ERASE/PROGRAM CYCLING

EEPROM cycling failures have always concerned users. The high electrical field required by thin oxide EEPROMs for tunneling can literally tear apart the oxide at defect regions. To combat this, some suppliers have implemented redundancy schemes, reducing cycling failures to insignificant levels. However, redundancy requires that cell size be doubled—an expensive solution.

Intel has designed extended cycling capability into its ETOX-II flash memory technology. Resulting improvements in cycling reliability come without increasing memory cell size or complexity. First, an advanced tunnel oxide increases the charge-carrying ability ten-fold. Second, the oxide area per cell subjected to the tunneling electric field is one-tenth that of common EEPROMs, minimizing the probability of oxide defects in the region. Finally, the peak electric field during erasure is approximately 2 mV/cm lower than EEPROM. The lower electric field greatly reduces oxide stress and the probability of failure—increasing time to wearout by a factor of 100,000,000.

The device is programmed and erased using Intel's Quick-Pulse Programming and Quick-Erase algorithms. Intel's algorithmic approach uses a series of operations (pulses), along with byte verification, to completely and reliably erase and program the device.

For further reliability information, see Reliability Report RR-60 (ETOX II Reliability Data Summary).

QUICK-PULSE PROGRAMMING ALGORITHM

The Quick-Pulse Programming algorithm uses programming operations of 10 μs duration. Each operation is followed by a byte verification to determine when the addressed byte has been successfully programmed. The algorithm allows for up to 25 programming operations per byte, although most bytes verify on the first or second operation. The entire sequence of programming and byte verification is performed with V_{PP} at high voltage. Figure 5 illustrates the Quick-Pulse Programming algorithm.

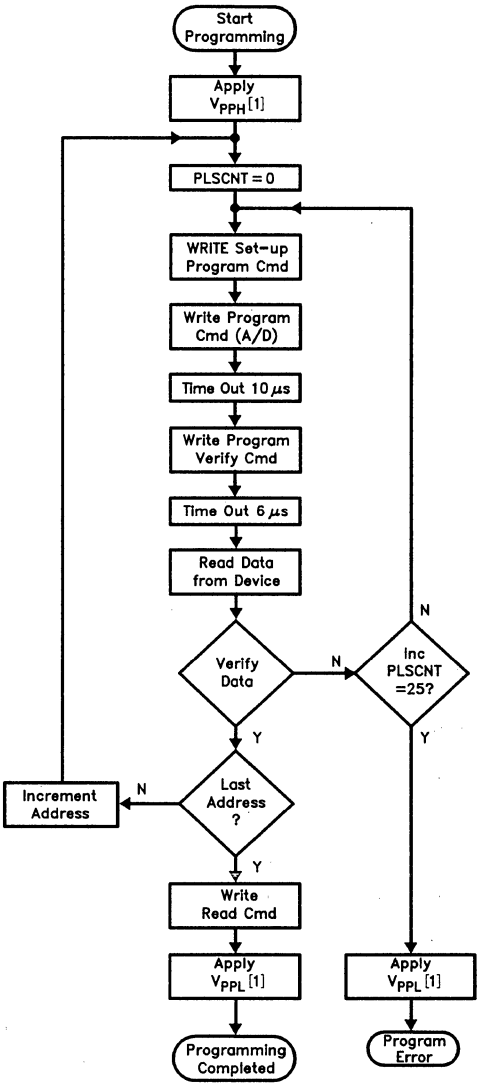
QUICK-ERASE ALGORITHM

Intel's Quick-Erase algorithm yields fast and reliable electrical erasure of memory contents. The algorithm employs a closed-loop flow, similar to the Quick-Pulse Programming algorithm, to simultaneously remove charge from all bits in the array.

Erasure begins with a read of memory contents. The 28F256A is erased when shipped from the factory. Reading FFH data from the device would immediately be followed by device programming.

Uniform and reliable erasure is ensured by first programming all bits in the device to their charged state (Data = 00H). This is accomplished, using the Quick-Pulse Programming algorithm, in approximately, one-half second.

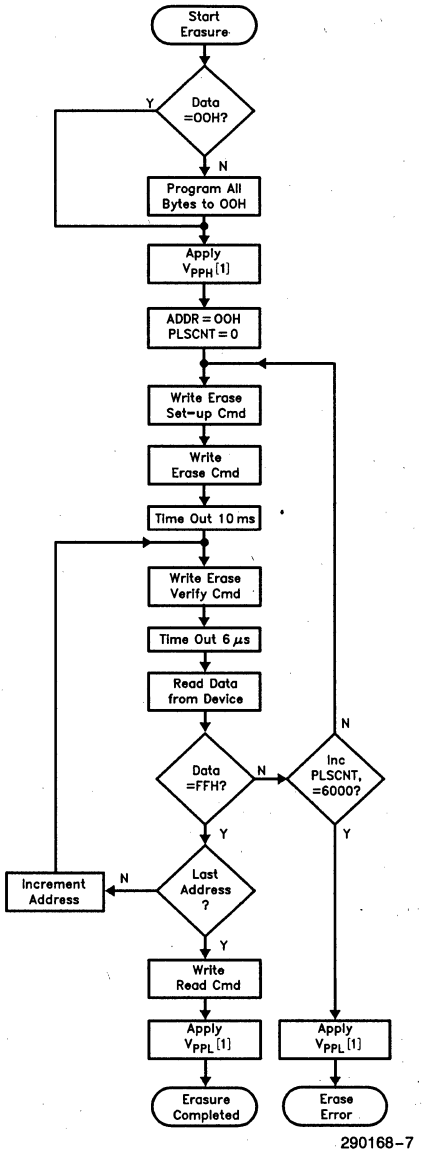
Erase execution then continues with an initial erase operation. Erase verification (data = FFH) begins at address 0000H and continues through the array to the last address, or until data other than FFH is encountered. With each erase operation, an increasing number of bytes verify to the erased state. Erase efficiency may be improved by storing the address of the last byte verified in a register. Following the next erase operation, verification starts at that stored address location. Erasure typically occurs in one second. Figure 6 illustrates the Quick-Erase Algorithm.



Bus Operation	Command	Comments
Standby		Wait for V_{PP} Ramp to V_{PPH}(1)
		Initialize Pulse-Count
Write	Set-up Program	Data = 40H
Write	Program	Valid Address/Data
Standby		Duration of Program Operation (t_{WHWH1})
Write	Program(3) Verify	Data = C0H; Stops Program Operation(2)
Standby		t_{WHGL}
Read		Read Byte to Verify Programming
Standby		Compare Data Output to Data Expected
Write	Read	Data = 00H, Resets the Register for Read Operations
Standby		Wait for V_{PP} Ramp to V_{PPL}(1)

NOTES:
1. See DC Characteristics for value of V_{PPH}. The V_{PP} power supply can be hard-wired to the device or switchable. When V_{PP} is switched, V_{PPL} may be ground, no-connect with a resistor tied to ground, or less than 6.5V. Refer to Principles of Operation.
2. Refer to Principles of Operation.
3. Program Verify is only performed after byte programming. A final read/compare may be performed (optional) after the register is written with the Read command.
4. CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.

Figure 5. 28F256A Quick-Pulse Programming Algorithm



290168-7

Bus Operation	Command	Comments
		Entire memory must = 00H before erasure
		Use Quick-Pulse Programming Algorithm (Figure 5)
Standby		Wait for V_{PP} Ramp to V_{PPH}(1)
		Initialize Addresses and Pulse-Count
Write	Set-up Erase	Data = 20H
Write	Erase	Data = 20H
Standby		Duration of Erase Operation (t_{WHWH2})
Write	Erase(3) Verify	Addr = Byte to Verify; Data = A0H; Stops Erase Operation(2)
Standby		t_{WHGL}
Read		Read Byte to Verify Erasure
Standby		Compare Output to FFH Increment Pulse-Count
Write	Read	Data = 00H, Resets the Register for Read Operations
Standby		Wait for V_{PP} Ramp to V_{PPL}(1)

NOTES:

1. See DC Characteristics for value of V_{PPH}. The V_{PP} power supply can be hard-wired to the device or switchable. When V_{PP} is switched, V_{PPL} may be ground, no-connect with a resistor tied to ground, or less than 6.5V. Refer to Principles of Operation.
2. Refer to Principles of Operation.
3. Erase Verify is performed only after chip-erasure. A final read/compare may be performed (optional) after the register is written with the read command.
4. CAUTION: The algorithm MUST BE FOLLOWED to ensure proper and reliable operation of the device.

Figure 6. 28F256A Quick-Erase Algorithm

DESIGN CONSIDERATIONS

Two-Line Output Control

Flash memories are often used in larger memory arrays. Intel provides two read-control inputs to accommodate multiple memory connections. Two-line control provides for:

a. the lowest possible memory power dissipation and,
b. complete assurance that output bus contention will not occur.

To efficiently use these two control inputs, an address-decoder output should drive chip-enable, while the system's read signal controls all flash memories and other parallel memories. This assures that only enabled memory devices have active outputs, while deselected devices maintain the low power standby condition.

Power Supply Decoupling

Flash memory power-switching characteristics require careful device decoupling. System designers are interested in three supply current (I_{CC}) issues—standby, active, and transient current peaks produced by falling and rising edges of chip-enable. The capacitive and inductive loads on the device outputs determine the magnitudes of these peaks.

Two-line control and proper decoupling capacitor selection will suppress transient voltage peaks. Each device should have a 0.1 μF ceramic capacitor connected between V_{CC} and V_{SS}, and between V_{PP} and V_{SS}.

Place the high-frequency, low-inherent-inductance capacitors as close as possible to the devices. Also, for every eight devices, a 4.7 μF electrolytic capacitor should be placed at the array's power supply connection, between V_{CC} and V_{SS}. The bulk capacitor will overcome voltage slumps caused by printed-circuit-board trace inductance, and will supply charge to the smaller capacitors as needed.

V_{PP} Trace on Printed Circuit Boards

Programming flash memories, while they reside in the target system, requires that the printed circuit board designer pay attention to the V_{PP} power supply trace. The V_{PP} pin supplies the memory cell current for programming. Use similar trace widths and layout considerations given the V_{CC} power bus. Adequate V_{PP} supply traces and decoupling will decrease V_{PP} voltage spikes and overshoots.

Power Up/Down Protection

The 28F256A is designed to offer protection against accidental erasure or programming, caused by spurious system-level signals that may exist during power transitions. Also, with its control register architecture, alteration of memory contents only occurs after successful completion of the two-step command sequences. Power supply sequencing is not required. Internal circuitry of the 28F256A ensures that the command register architecture is reset to the read mode on power up.

A system designer must guard against active writes for V_{CC} voltages above V_{LKO} when V_{PP} is active. Since both WE# and CE# must be low for a command write, driving either to V_{IH} will inhibit writes. The control register architecture provides an added level of protection since alteration of memory contents only occurs after successful completion of the two-step command sequences.

28F256A Power Dissipation

When designing portable systems, designers must consider battery power consumption not only during device operation, but also for data retention during system idle time. Flash memory nonvolatility increases the usable battery life of your system because the 28F256A does not consume any power to retain code or data when the system is off. Table 4 illustrates the power dissipated when updating the 28F256A.

Table 4. 28F256A Typical Update Power Dissipation(4)

Operation	Power Dissipation (Watt-Seconds)	Notes
Array Program/Program Verify	0.043	1
Array Erase/Erase Verify	0.083	2
One Complete Cycle	0.169	3

NOTES:
1. Formula to calculate typical Program/Program Verify Power = [$V_{PP} \times$ # Bytes $\times$ typical # Prog Pulses ($t_{WHWH1} \times I_{PP2}$ typical + $t_{WHGL} \times I_{PP4}$ typical)] + [$V_{CC} \times$ # Bytes $\times$ typical # Prog Pulses ($t_{WHWH1} \times I_{CC2}$ typical + $t_{WHGL} \times I_{CC2}$ typical)].
2. Formula to calculate typical Erase/Erase Verify Power = [V_{PP} (I_{PP3} typical $\times$ t_{ERASE} typical + I_{PP5} typical $\times$ $t_{WHGL} \times$ # Bytes)] + [V_{CC}(I_{CC3} typical $\times$ t_{ERASE} typical + I_{CC5} typical $\times$ $t_{WHGL} \times$ # Bytes)].
3. One Complete Cycle = Array Preprogram + Array Erase + Program.
4. "Typicals" are not guaranteed, but based on a limited number of samples from production lots.

ABSOLUTE MAXIMUM RATINGS*

Operating Temperature
During Read ..... −40°C to +125°C(1)
During Erase/Program ..... −40°C to +125°C

Temperature Under Bias ..... −40°C to +125°C

Storage Temperature ..... −65°C to +150°C

Voltage on Any Pin with
Respect to Ground ..... −2.0V to +7.0V(2)

Voltage on Pin A_9 with
Respect to Ground ..... −2.0V to +13.5V(2, 3)

V_{PP} Supply Voltage with
Respect to Ground
During Erase/Program .... −2.0V to +14.0V(2, 3)

V_{CC} Supply Voltage with
Respect to Ground ..... −2.0V to +7.0V(2)

Output Short Circuit Current ..... 100 mA(4)

Maximum Junction Temperature (T_J) ..... 140°C

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

NOTES:
1. Operating temperature is for automotive product defined by this specification.
2. Minimum DC input voltage is −0.5V. During transitions, inputs may undershoot to −2.0V for periods less than 20 ns. Maximum DC voltage on output pins is V_{CC} + 0.5V, which may overshoot to V_{CC} + 2.0V for periods less than 20 ns.
3. Maximum DC voltage on A_9 or V_{PP} may overshoot to +14.0V for periods less than 20 ns.
4. Output shorted for no more than one second. No more than one output shorted at a time.

OPERATING CONDITIONS

Symbol	Parameter	Limits		Unit	Comments
		Min	Max		
T_A	Operating Temperature	−40	+125	°C	For Read-Only and Read/Write Operations
V_{CC}	V_{CC} Supply Voltage	4.50	5.50	V	

DC CHARACTERISTICS—TTL/NMOS COMPATIBLE

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1			±1.0	µA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1			±10	µA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1			1.0	mA	$V_{CC} = V_{CC}$ Max CE# = V_{IH}
I_{CC1}	V_{CC} Active Read Current	1		10	30	mA	$V_{CC} = V_{CC}$ Max, CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Programming Current	1, 2		1.0	30	mA	Programming in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		5.0	30	mA	Erasure in Progress
I_{CC4}	V_{CC} Program Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
I_{PPS}	V_{PP} Leakage Current	1			±10	µA	$V_{PP} \leq V_{CC}$

8

DC CHARACTERISTICS—TTL/NMOS COMPATIBLE (Continued)

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{PP1}	V_{PP} Read Current or Standby Current	1		90	200	µA	$V_{PP} > V_{CC}$
					±10	µA	$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Programming Current	1, 2		8.0	30	mA	$V_{PP} = V_{PPH}$ Programming in Progress
I_{PP3}	V_{PP} Erase Current	1, 2		4.0	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Program Verify Current	1, 2		2.0	5.0	µA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		2.0		$V_{CC} + 0.5$	V	
V_{OL}	Output Low Voltage				0.45	V	$I_{OL} = 2.1$ mA $V_{CC} = V_{CC}$ Min
V_{OH1}	Output High Voltage		2.4			V	$I_{OH} = -2.5$ mA $V_{CC} = V_{CC}$ Min
V_{ID}	A_9 Intelligent Identifer Voltage		11.50		13.00	V	$A_9 = V_{ID}$
I_{ID}	V_{CC} ID Current	1		10	30	mA	$A_9 = V_{ID}$
	V_{PP} ID Current			90	500	µA	
V_{PPL}	V_{PP} during Read-Only Operations		0.00		6.5	V	**NOTE:** Erase/Program are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} during Read/Write Operations		11.40		12.60	V	
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

DC CHARACTERISTICS—CMOS COMPATIBLE

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{LI}	Input Leakage Current	1			±1.0	µA	$V_{CC} = V_{CC}$ Max $V_{IN} = V_{CC}$ or V_{SS}
I_{LO}	Output Leakage Current	1			±10	µA	$V_{CC} = V_{CC}$ Max $V_{OUT} = V_{CC}$ or V_{SS}
I_{CCS}	V_{CC} Standby Current	1		50	100	µA	$V_{CC} = V_{CC}$ Max CE# = V_{CC} ±0.2V
I_{CC1}	V_{CC} Active Read Current	1		10	30	mA	$V_{CC} = V_{CC}$ Max, CE# = V_{IL} f = 6 MHz, I_{OUT} = 0 mA
I_{CC2}	V_{CC} Programming Current	1, 2		1.0	30	mA	Programming in Progress
I_{CC3}	V_{CC} Erase Current	1, 2		5.0	30	mA	Erasure in Progress
I_{CC4}	V_{CC} Program Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{CC5}	V_{CC} Erase Verify Current	1, 2		5.0	30	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress

DC CHARACTERISTICS—CMOS COMPATIBLE (Continued)

Symbol	Parameter	Notes	Limits			Unit	Test Conditions
			Min	Typical	Max		
I_{PPS}	V_{PP} Leakage Current	1			±10	μA	$V_{PP} \leq V_{CC}$
I_{PP1}	V_{PP} Read Current or Standby Current	1		90	200	μA	$V_{PP} > V_{CC}$
					±10		$V_{PP} \leq V_{CC}$
I_{PP2}	V_{PP} Programming Current	1, 2		8.0	30	mA	$V_{PP} = V_{PPH}$ Programming in Progress
I_{PP3}	V_{PP} Erase Current	1, 2		4.0	30	mA	$V_{PP} = V_{PPH}$ Erasure in Progress
I_{PP4}	V_{PP} Program Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Program Verify in Progress
I_{PP5}	V_{PP} Erase Verify Current	1, 2		2.0	5.0	mA	$V_{PP} = V_{PPH}$ Erase Verify in Progress
V_{IL}	Input Low Voltage		−0.5		0.8	V	
V_{IH}	Input High Voltage		0.7 V_{CC}		V_{CC} + 0.5	V	
V_{OL}	Output Low Voltage				0.45	V	I_{OL} = 2.1 mA $V_{CC} = V_{CC}$ Min
V_{OH1}	Output High Voltage		0.85 V_{CC}			V	I_{OH} = −2.5 mA, $V_{CC} = V_{CC}$ Min
V_{OH2}			V_{CC} − 0.4				I_{OH} = −100 μA, $V_{CC} = V_{CC}$ Min
V_{ID}	A_9 Intelligent Identifer Voltage		11.50		13.00	V	
I_{ID}	V_{CC} ID Current	1		10	30	mA	A_9 = ID
I_{ID}	V_{PP} ID Current	1		90	500	μA	A_9 = ID
V_{PPL}	V_{PP} during Read-Only Operations		0.00		6.5	V	**NOTE:** Erase/Programs are Inhibited when $V_{PP} = V_{PPL}$
V_{PPH}	V_{PP} during Read/ Write Operations		11.40		12.60	V	V_{PP} = 12.0V
V_{LKO}	V_{CC} Erase/Write Lock Voltage		2.5			V	

CAPACITANCE[3] T_A = 25°C, f = 1.0 MHz

Symbol	Parameter	Limits		Unit	Conditions
		Min	Max		
C_{IN}	Address/Control Capacitance		8	pF	V_{IN} = 0V
C_{OUT}	Output Capacitance		12	pF	V_{OUT} = 0V

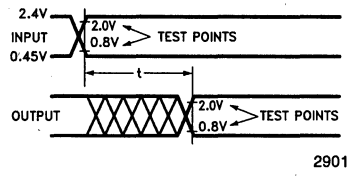
NOTES:
1. All currents are in RMS unless otherwise noted. Typical values at V_{CC} = 5.0V, V_{PP} = 12.0V, T = 25°C.
2. Not 100% tested: characterization data available.
3. Sampled, not 100% tested.
4. "Typicals" are not guaranteed, but are based on a limited number of samples from production lots.

8

AC TEST CONDITIONS

Input Rise and Fall Times (10% to 90%) ..... 10 ns
Input Pulse Levels ..... 0.45V and 2.4V
Input Timing Reference Level ..... 0.8V and 2.0V
Output Timing Reference Level ..... 0.8V and 2.0V

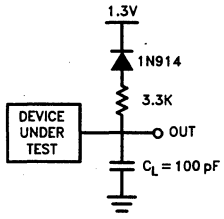
AC TESTING INPUT/OUTPUT WAVEFORM



290168-8

AC Testing: Inputs are driven at 2.5V for a logic "1" and 0.45V for a logic "0". Testing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0". Rise/Fall time ≤ 10 ns.

AC TESTING LOAD CIRCUIT



290168-9

C_L = 100 pF
C_L includes Jig Capacitance

AC CHARACTERISTICS—Read-Only Operations(2)

Versions			28F256A-120		28F256A-150		Unit
Symbol	Characteristic	Notes	Min	Max	Min	Max	
t_{AVAV}/t_{RC}	Read Cycle Time	3	120		150		ns
t_{ELQV}/t_{CE}	Chip Enable Access Time			120		150	ns
t_{AVQV}/t_{ACC}	Address Access Time			120		150	ns
t_{GLQV}/t_{OE}	Output Enable Access Time			50		55	ns
t_{ELQX}/t_{LZ}	Chip Enable to Output in Low Z	3	0		0		ns
t_{EHQZ}	Chip Disable to Output in High Z	3		50		55	ns
t_{GLQX}/t_{OLZ}	Output Enable to Output in Low Z	3	0		0		ns
t_{GHQZ}/t_{DF}	Output Disable to Output in High Z	4		30		35	ns
t_{OH}	Output Hold from Address, CE#, or OE# Change	1, 3	0		0		ns
t_{WHGL}	Write Recovery Time before Read		6		6		μs

NOTES:
1. Whichever occurs first.
2. Rise/Fall Time ≤ 10 ns.
3. Not 100% tested: characterization data available.
4. Guaranteed by design.

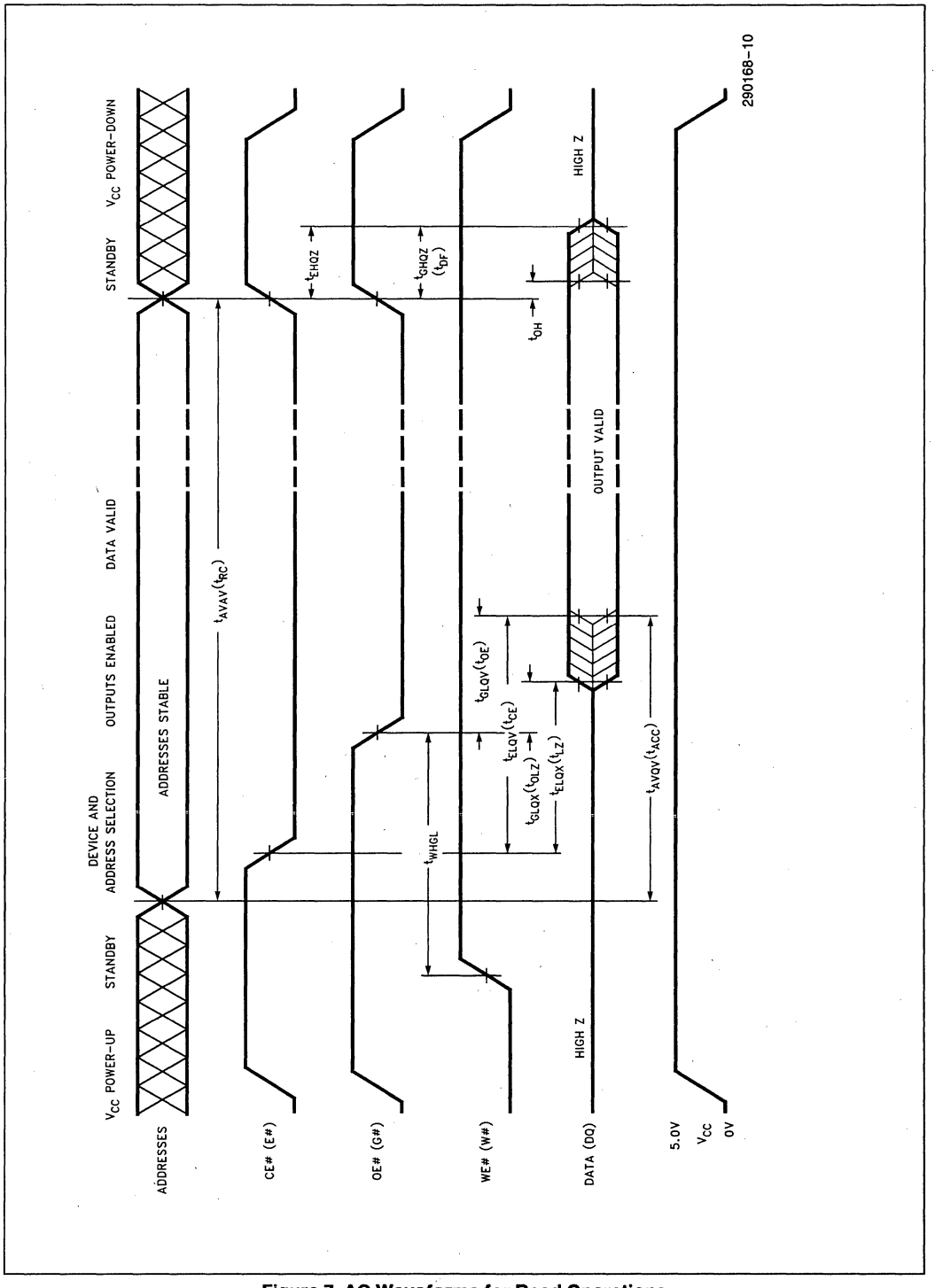


Figure 7. AC Waveforms for Read Operations

8

AC CHARACTERISTICS—Write/Erase/Program Operations[1, 3]

Versions			28F256A-120		28F256A-150		Unit
Symbol	Characteristic	Notes	Min	Max	Min	Max	
t_{AVAV}/t_{WC}	Write Cycle Time		120		150		ns
t_{AVWL}/t_{AS}	Address Set-Up Time		0		0		ns
t_{WLAX}/t_{AH}	Address Hold Time	2	60		60		ns
t_{DVWH}/t_{DS}	Data Set-up Time		50		50		ns
t_{WHDX}/t_{DH}	Data Hold Time		10		10		ns
t_{WHGL}	Write Recovery Time before Read		6		6		μs
t_{GHWL}	Read Recovery Time before Write		0		0		μs
t_{ELWL}/t_{CS}	Chip Enable Set-Up Time before Write	2	20		20		ns
t_{WHEH}/t_{CH}	Chip Enable Hold Time		0		0		ns
t_{WLWH}/t_{WP}	Write Pulse Width	2	80		80		ns
t_{ELEH}	Alternative Write Pulse Width	2	80		80		ns
t_{WHWL}/t_{WPH}	Write Pulse Width High		20		20		ns
t_{WHWH1}	Duration of Programming Operation	4	10		10		μs
t_{WHWH2}	Duration of Erase Operation	4	9.5		9.5		ms
t_{VPEL}	V_{PP} Set-Up Time to Chip Enable Low		1.0		1.0		ms

NOTES:
1. Read timing characteristics during read/write operations are the same as during read-only operations. Refer to AC Characteristics for Read-Only Operations.
2. Chip-Enable Controlled Writes: Write operations are driven by the valid combination of Chip-Enable and Write-Enable. In systems where Chip-Enable defines the write pulse width (within a longer Write-Enable timing waveform) all set-up, hold, and inactive Write-Enable times should be measured relative to the Chip-Enable waveform.
3. Rise/Fall time ≤ 10 ns.
4. The integrated stop timer terminates the programming/erase operations, thereby eliminating the need for a maximum specification.

ERASE AND PROGRAMMING PERFORMANCE

Parameter	Notes	Limits			Unit	Comments
		Min	Typ	Max		
Chip Erase Time	1, 3, 4, 6		1	60	Sec	Excludes 00H Programming Prior to Erasure
Chip Program Time	1, 2, 4		0.5	3.1	Sec	Excludes System-Level Overhead
Erase/Program Cycles	1, 3, 5	1,000	100,000		Cycles	

NOTES:
1. "Typicals" are not guaranteed, but based on a limited number of samples taken from production lots. Data taken at $T = 25°C$, $V_{PP} = 12.0V$, $V_{CC} = 5.0V$.
2. Minimum byte programming time excluding system overhead is 16 μs (10 μs program + 6 μs write recovery), while maximum is 400 μs/byte (16 μs × 25 loops allowed by algorithm). Max chip programming time is specified lower than the worst case allowed by the programming algorithm since most bytes program significantly faster than the worst case byte.
3. Excludes 00H Programming prior to erasure.
4. Excludes system-level overhead.
5. Refer to RR-60 "ETOX Flash Memory Reliability Data Summary" for typical cycling data and failure rate calculations.
6. Maximum erase specification is determined by algorithmic limit and accounts for cumulative effect of erasure at $T = -40°C$, 1,000 cycles, $V_{PP} = 11.4V$, $V_{CC} = 4.5V$.

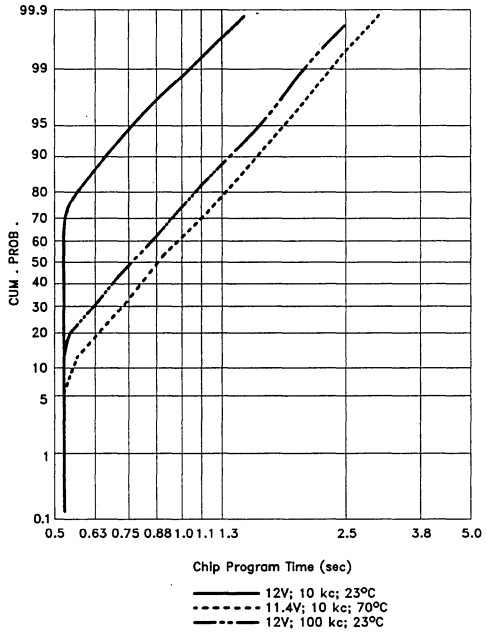


Figure 8. 28F256A Typical Programming Capability
See Note 1, Page 19

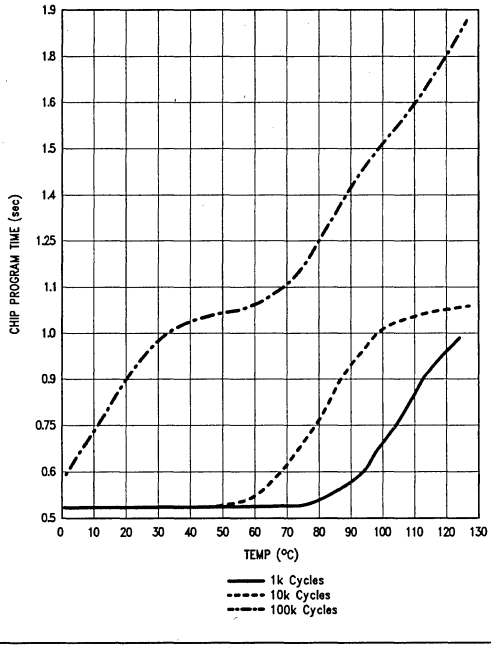


Figure 9. 28F256A Typical Program Time at V_{PP} = 12.0V

8

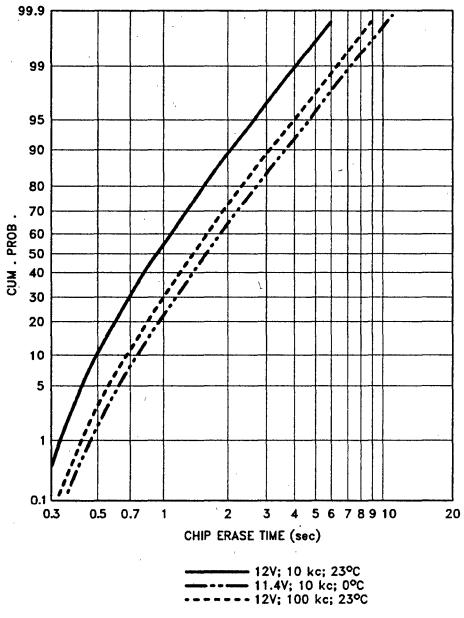


Figure 10. 28F256A Typical Erase Capability
See Note 1, Page 19

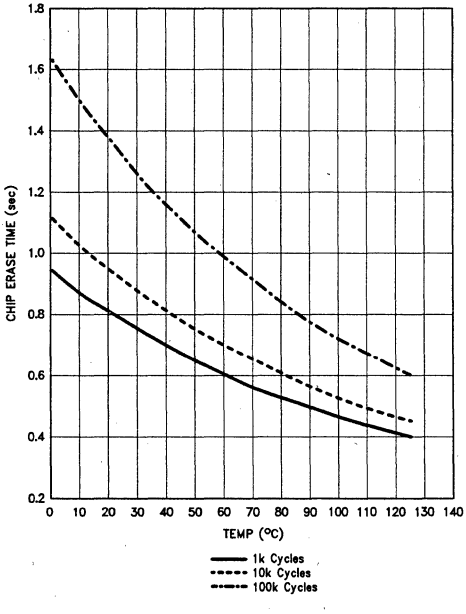
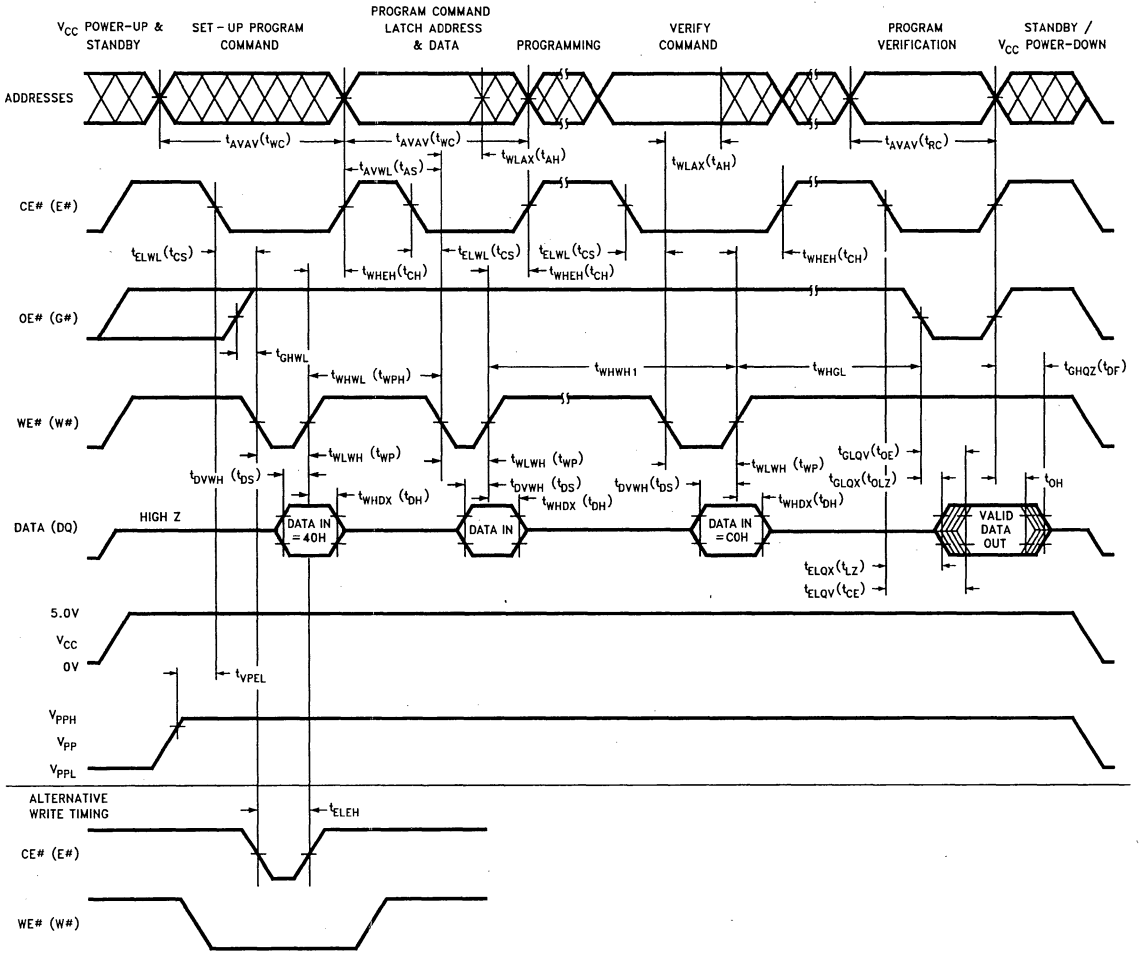


Figure 11. 28F256A Typical Erase Time at V_{PP} = 12.0V



290168-11

Figure 12. AC Waveforms for Programming Operations

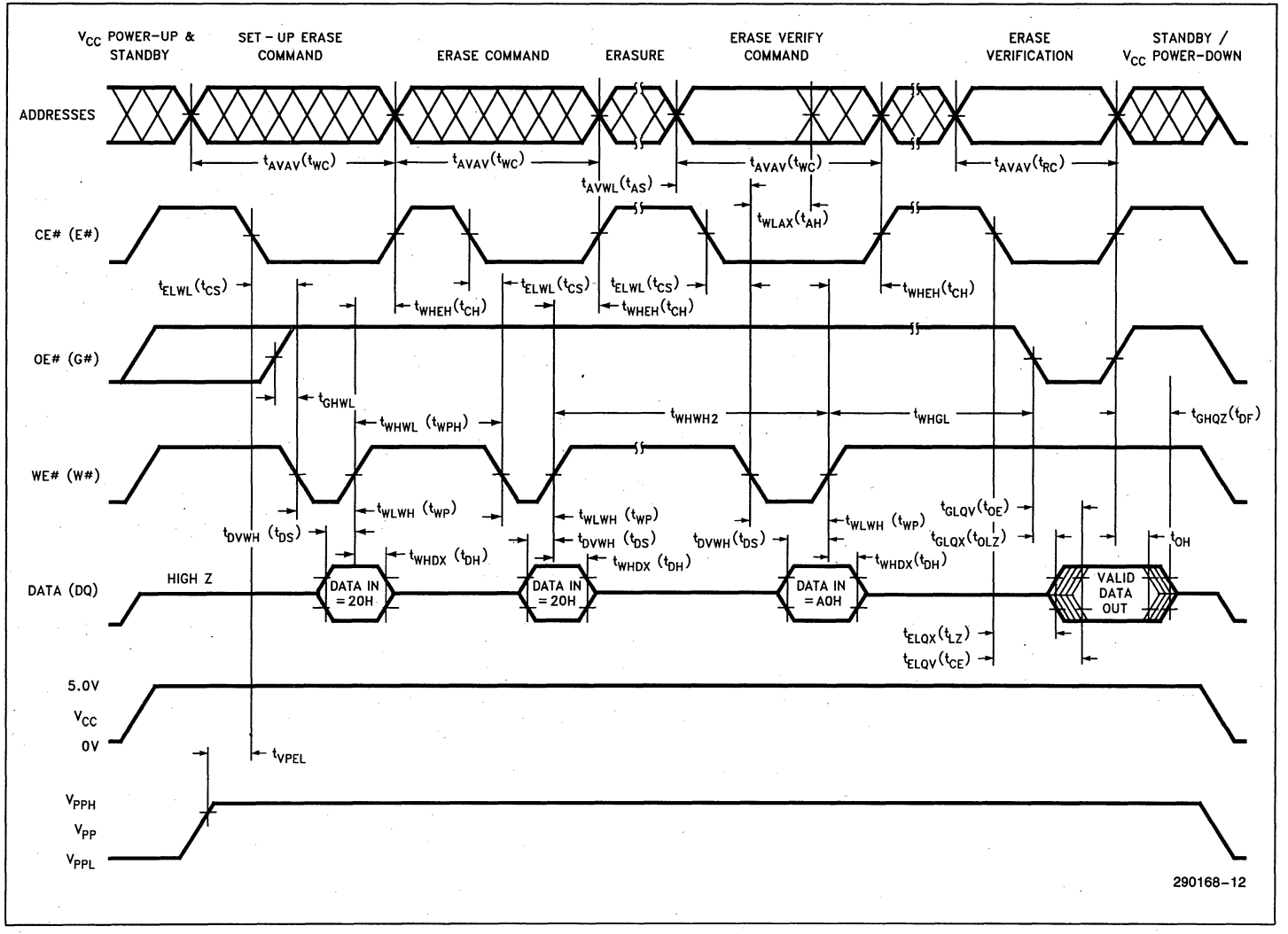
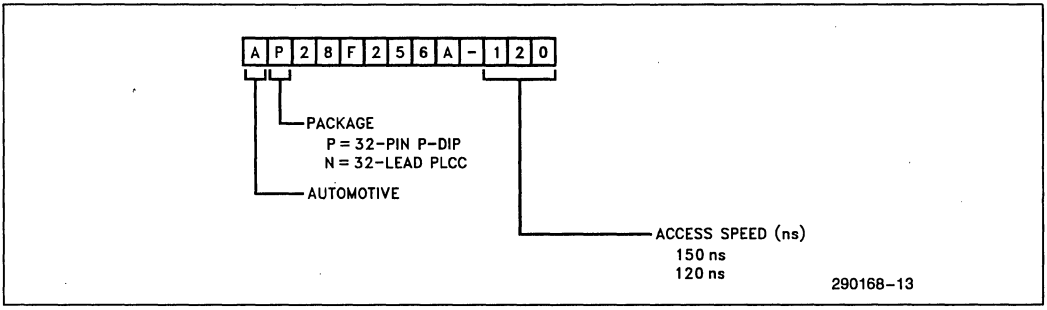


Figure 13. AC Waveforms for Erase Operations

Ordering Information



VALID COMBINATIONS:

AP28F256A-120	AP28F256A-120
AP28F256A-150	AN28F256A-150

ADDITIONAL INFORMATION

	Order Number
AP-316, "Using the 28F256A Flash Memory for In-System Reprogrammable Non-Volatile Storage"	292046
ER-21, "The Intel 28F256 Flash Memory"	294004
RR-60, "ETOX™ II Flash Memory Reliability Data Summary"	293002
AP-325, "Guide to Flash Memory Reprogramming"	292059

REVISION HISTORY

Number	Description
005	Changed Erase/Program Cycles to 1,000 minimum.
006	Added 120 ns Speed

8

intel®

9

Process Engineering Reports

intel®

ER-20

ENGINEERING REPORT

ETOX™ II Flash Memory Technology

JASON ZILLER
PRODUCT ENGINEERING

9

September 1993

Order Number: 294005-006

ETOX™ II FLASH MEMORY TECHNOLOGY

CONTENTS

PAGE

INTRODUCTION 9-3

ETOX™ II FLASH MEMORY CELL 9-3

MEMORY ARRAY CONSIDERATIONS 9-3

ETOX™ II FLASH MEMORY RELIABILITY 9-3

SUMMARY 9-4

INTRODUCTION

Intel's ETOX™ II (EPROM tunnel oxide) flash memory technology is derived from the CHMOS** III-E EPROM technology. It replaces ultraviolet erasability with a non-volatile memory cell that is electrically erasable in bulk array form. Intel flash memory combines the EPROM programming mechanism with EEPROM erase, producing a versatile memory device that is highly reliable and cost effective. This report describes the fundamentals of the ETOX II flash memory cell in comparison to the standard EPROM, and gives insight into its operation in a system environment.

The ETOX II flash memory cell is nearly identical in size to CHMOS III-E EPROM. This allows comparable densities. The primary difference between ETOX II flash memory and EPROM cells is the flash memory cell's thinner gate oxide, which permits the electrical erase capability. (See Photo 1.)

ETOX II FLASH MEMORY CELL

Intel's ETOX II flash memory cell is composed of a single transistor with a floating gate for charge storage, like the traditional EPROM. (See Figure 1.) In contrast, conventional two-transistor EEPROM cells are typically much larger. Intel produces ETOX II flash memory devices on 1.0μ photolithography.

The ETOX II cell's programming mechanism is identical to the EPROM; that is, hot channel electron injection. The device programming mode forces the cell's control gate and drain to a high voltage while leaving the source grounded. The high drain voltage generates "hot" electrons that are swept across the channel. These hot electrons collide with other atoms along the way, creating even more free electrons. Meanwhile, the high voltage on the control gate attracts these free electrons across the lower gate oxide into the floating gate, where they are trapped. (See Figure 2.) Typically, this process takes less than 10 μs.

Flash memory's advantage over EPROM is electrical erasure, discharging the floating gate without ultraviolet light exposure. The erase mechanism is an EEPROM adaptation which uses "Fowler-Nordheim"[1] tunneling. A high electric field across the lower gate oxide pulls electrons off the floating gate. The erase mode routes the same external voltage used for programming to the source of the memory cell, while the gate is grounded and the drain is left disconnected. (Figure 3.)

MEMORY ARRAY CONSIDERATIONS

The ETOX II flash memory cells have the same array configuration as standard EPROM, thereby matching EPROM in density. Also, identical peripheral circuitry for normal access achieves the same read performance as the Intel CHMOS III-E EPROMs.

Intel flash memory's programming circuitry is also identical to Intel's EPROM designs. Row decoders drive the selected wordline to high voltage, while input data combined with column decoders determine the number of bitlines that are gated to high voltage. This provides the same byte programmability as an EPROM. Intel Flash Memories offer the efficient Quick-Pulse Programming algorithm that is featured on advanced EPROMs.

Array erase is unique to flash memory technology. Unlike conventional EEPROMs, which use a select transistor for individual byte erase control, flash memories achieve much higher density with single transistor cells. Therefore, the erase mode supplies high voltage to the sources of every cell simultaneously, performing a full array erasure. A programming operation must be performed before every erase to equalize the amount of charge on each cell. Then Intel's Quick-Erase algorithm intelligently erases the array down to the appropriate minimum threshold level required to read all "ones" data. This procedure ensures a tight distribution of erased cell thresholds throughout the array.

ETOX II FLASH MEMORY RELIABILITY

The reliability of Intel's CHMOS ETOX II flash memory process is equivalent to its sister EPROM technology. The ETOX II and EPROM processes share the same data retention characteristics. Qualification data shows that 1 Megabit flash memories produced on the ETOX II process provide 100,000 program and erase cycles with no cycling failures due to oxide stress or breakdown. This extended cycling capability is attributed to improvements in tunnel oxide processing and advantages inherent in the ETOX II cell approach.

9

[1]M. Lenzlinger, E.H. Snow, "Fowler-Nordheim Tunneling into Thermally Grown SiO2," Journal of Applied Physics, Vol. 40 (1969), p. 278.

*Intel's ETOX II flash memory process has patents pending.

**CHMOS is a patented process of Intel Corporation.

SUMMARY

ETOX II flash memory technology is the optimal combination of EPROM and EEPROM technologies. Intel's ETOX II flash memory process offers extended cycling capability with the density and manufacturability of EPROMs. From an application standpoint, flash memory technology provides the capability to improve overall system quality throughout the product development and manufacturing stages. Also, flash memory density is ideally suited for applications requiring version updates of entire programs which, in turn, suit the "flash" characteristics of erasing the entire array at once. In addition, individual byte programming allows for data acquisition. Flash memory devices produce on the ETOX II process provide a high density, low cost solution to many system memory storage requirements which were previously unavailable.

Table I

	EPROM	ETOX II Flash Memory	EEPROM
Normalized Cell Size	1.0	1.2–1.3	3.0
Programming:			
Mechanism	Hot Electron Injection	Hot Electron Injection	Tunneling Byte
Resolution	Byte	Byte	5 ms
Typ. Time	< 100 μs	< 10 μs	
Erase:			
Mechanism	UV Light	Tunneling	Tunneling
Resolution	Bulk Array	Bulk Array	Byte
Typ. Time	20 Min.	< 1 Sec.	5 ms

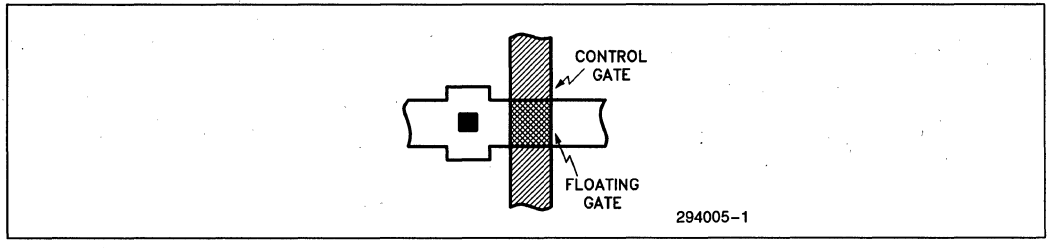


Figure 1. ETOX II Flash Memory Cell Layout (Top View)

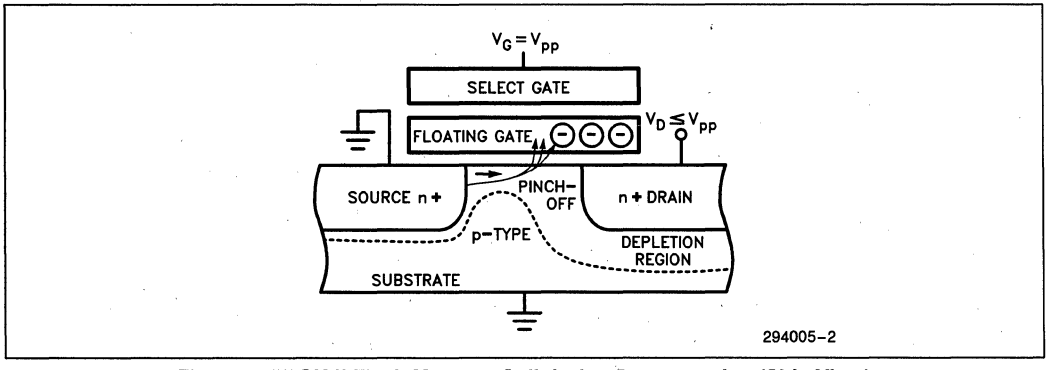


Figure 2. ETOX II Flash Memory Cell during Programming (Side View)

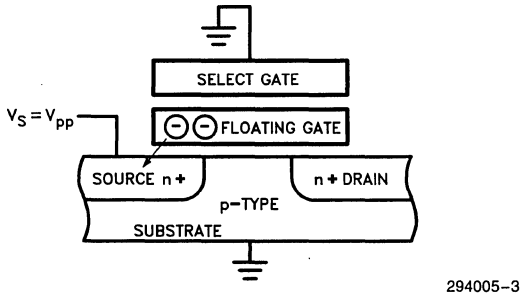
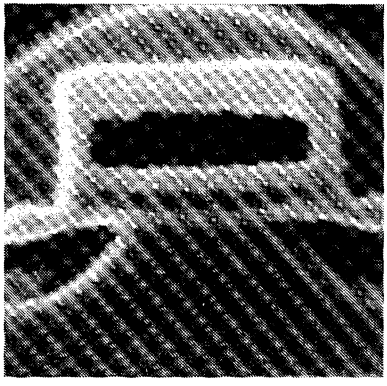


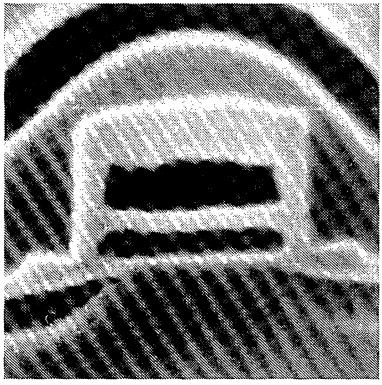
Figure 3. ETOX II Flash Memory Cell during Erase (Side View)

PHOTO 1



294005-4

ETOX II Cell
(50,000 x Magnification)



294005-5

CHMOS III-E EPROM Cell
(50,000 x Magnification)

intel®

ER-28

ENGINEERING REPORT

ETOX™ III Flash Memory Technology

ALAN BUCHECKER
MARK NEWMAN
MEMORY COMPONENTS DIVISION

March 1994

Order Number: 294012-002

ETOX™ III Flash Memory Technology

CONTENTS

PAGE

INTRODUCTION 9-8

ETOX™ III FLASH MEMORY CELL 9-8

Cell Processing 9-9

Byte Write 9-9

Block Erase 9-10

ETOX III PROCESS CHARACTERISTICS 9-10

Write/Erase Performance with Voltage and Temperature 9-10

Write/Erase Cycling 9-11

Process Variation 9-12

Electrical Testing 9-12

CONTENTS

PAGE

ETOX III MEMORY ARRAY CONSIDERATIONS 9-12

28F008SA Array Architecture 9-12

28F008SA Byte-Write Operation 9-12

28F008SA Block-Erase Operation 9-12

28F008SA Cell Voltage Threshold 9-12

FLASH VS. OTHER SEMICONDUCTOR MEMORY TECHNOLOGIES 9-13

SRAM and DRAM 9-15

EPROM 9-16

EEPROM 9-16

NAND 9-17

SUMMARY 9-17

OTHER REFERENCES 9-17

APPENDIX A 9-18

INTRODUCTION

Intel's ETOX™ III (EPROM tunnel oxide) flash memory technology builds on previous flash and EPROM processes spanning over two decades of manufacturing experience. This third-generation process produces devices based on 0.8 μm photolithography. Intel's Flash Memories combine EPROM programming with EEPROM-like in-system electrical erasure. This functionality, experience and technology yield a versatile non-volatile memory that is highly reliable and cost effective.

ETOX III cell integrity enables applications requiring 100,000 write/erase cycles. New designs incorporate array blocking schemes and on-chip automation of write and erase to simplify customer designs and software interface. These features combine with existing Intel Flash Memory technology to produce a device that can be termed a block-alterable non-volatile RAM. Access time (t_{ACC}) and die size decrease via this smaller photolithography, making new Intel Flash Memories competitive with DRAM in read speed and cost.

This report describes the fundamentals of Intel's ETOX III flash memory cell. It provides insight into device reliability and performance enhancements based on ETOX III advances, and compares other semiconductor memory technologies.

The Intel 28F008SA 8-Mbit flash memory is the first ETOX III product offering. This report references the 28F008SA to explain device-level concepts, and ends by highlighting important flash memory application trends.

ETOX III FLASH MEMORY CELL

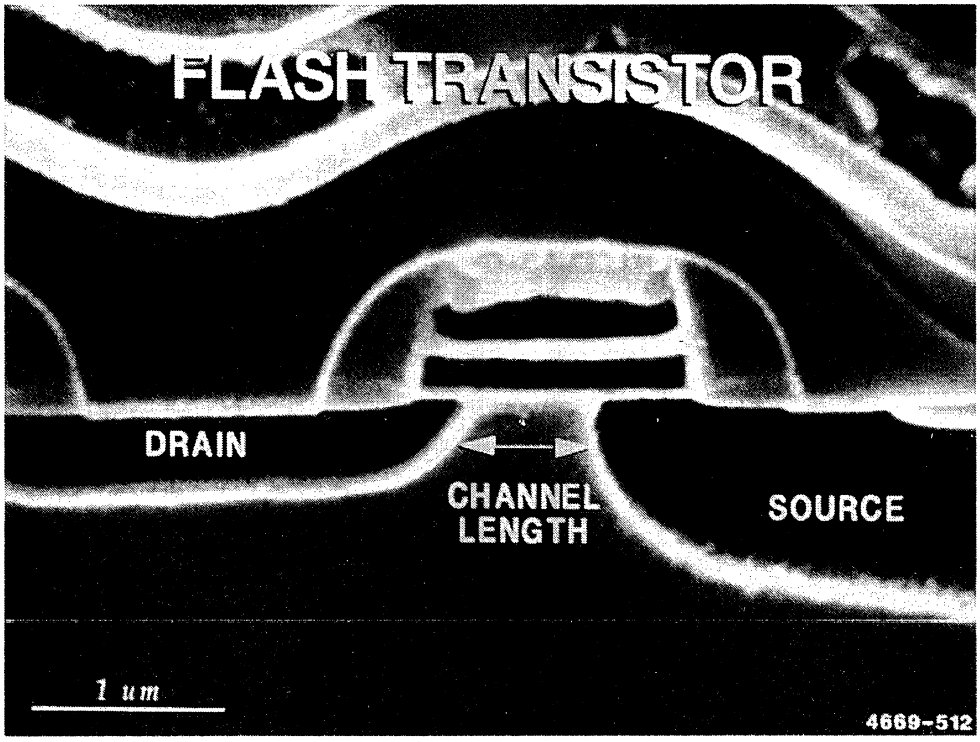
ETOX III is a 0.8 μm double-polysilicon N-well and P-well CMOS process. This lithographic advance improves memory cell/array compaction more than two-fold over its predecessor, the 1.0 μm ETOX II process. ETOX III-aided compaction allows for a 4x product density growth given current packaging constraints.

Second-generation device architecture (see Appendix A) and 0.8 μm geometries increase byte-write and read access performance by 2x over ETOX II products. Double-metal technology enhances these improvements by aiding die size compaction, and wordline strapping. EPI wafer processing, which reduces latch-up, also factors into this performance boost by shrinking transistor layout.

Cell Processing

Intel's single-transistor Flash Memory cell stores charge on a floating polysilicon gate. Dimensions of 2.5 μm by 2.9 μm make an ETOX III cell measuring 7.25 μm^2. Cell layout locates the polysilicon control gate above the floating gate (Figure 1). Tungsten silicide, deposited on the control gate, reduces wordline resistance. Two dielectrics isolate metal-1 from the control gate.

Inter-poly dielectrics of oxide and nitride isolate the floating gate from the control gate. A very thin tunnel oxide (~ 100Å) separates the floating gate from its silicon interface. Both the floating and control gates have additional isolation between them and their respective source/drain regions. A deeper source diffusion prevents breakdown during erase operations. In the array metal-2 straps the wordline to enhance access times. As with ETOX I and ETOX II, metal-1 carries bitline data to the sense amps and routes voltages to cell sources.



294012-18

Figure 1. ETOX III Flash Memory Cell (Side View)

Byte Write

Writing data to an addressed byte transitions selected cells from the "1" (erased or no charge) state to the "0" (charged) state. This involves a programming mechanism called channel hot-electron injection. When programming (Figure 2), a cell's control gate (wordline) connects to the external programming supply voltage (V_{PP} at 12V). The drain (bitline) sees an intermediate level (~ $V_{PP}/2$), while the source is at ground. V_{PP} on the control gate capacitively couples to the floating gate through the intervening dielectric. This coupling raises the floating gate to a programming voltage, inverting the channel underneath.

The channel electrons now have a higher drift velocity, with resulting increased kinetic energy. Collisions between these electrons and substrate atoms heat the silicon lattice. At the programming bias voltage, the electrons cannot transfer their kinetic energy to the atoms fast enough to maintain a thermal balance. They become "hotter," and many scatter toward the tunnel oxide. These electrons overcome the 3.1eV (electron voltage) tunnel oxide barrier and accumulate onto the floating gate.

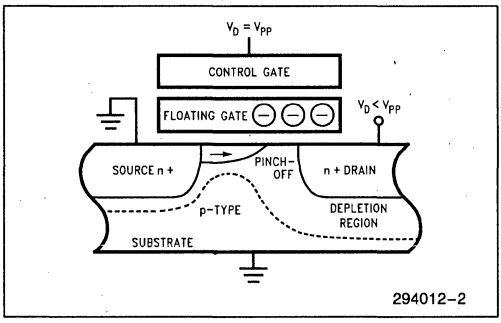


Figure 2. ETOX III Flash Memory Cell during Programming (Side View)

The electrons stored on the floating gate raise the turn-on voltage threshold (V_t) of that cell. During device read operations this transistor remains in the off state. A "0" results at the output because the "off" cell does not pass current, causing the bitline to electrically stay at/pull-up to the V_{CC} read voltage.

The internal programming pulse on ETOX III is 4 μs (excluding WSM overhead), reduced from 10 μs on ETOX II devices. This optimized pulse width yields faster byte-write times and greater cycling reliability. Like previous ETOX products, the automated WSM allows for the occasional byte requiring more than one pulse.

Block Erase

During erasure, electrons are pulled off selected memory cells simultaneously. The erase process ("Fowler-Nordheim" tunneling) starts by routing V_{PP} to the source, ground to the select gate, and floats the drain (Figure 3).

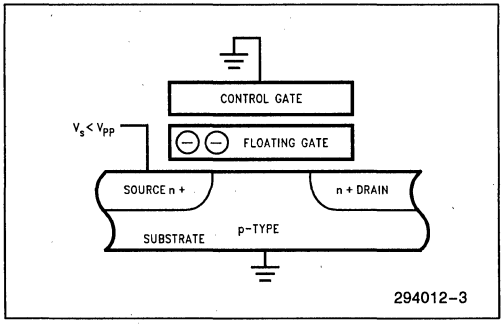


Figure 3. ETOX III Flash Memory Cell during Erase (Side View)

While biased in this fashion, electrons tunnel off the floating gate. They pass through the thin oxide to the source, lowering that cell's V_t. During a read operation, the resulting "1" at the output corresponds to an "on" cell discharging its bitline through the grounded source.

Erase automation sets the internal V_{PP} pulse to 10 ms. The WSM provides sufficient 10 ms pulses, and automatically verifies all memory cells in a given block. This optimized pulse width enhances block-erase time and cycling endurance.

ETOX III PROCESS CHARACTERISTICS

Intel leverages over two decades of EPROM/flash technology and manufacturing experience to produce reliable memory products. Refined processing techniques inherent to new Intel memory technologies and continuous improvements in process control ensure tunnel oxide quality. A scaled substrate EPI thickness reduces product latch-up. Double-metal technology requires improved planarization processing, which in turn enhances moisture performance. Additionally, decreases in defect density show lasting cell integrity in cycling and data retention.

Write/Erase Performance with Voltage and Temperature

Voltage affects byte-write and block-erase performance. Maximum V_{PP} improves byte-write and block-erase times. Figure 4 shows little difference in block-write time versus V_{PP}, but visible differences in block-erase performance. A secondary and negligible effect results from the operating supply voltage (V_{CC}). Byte-write and block-erase times are guaranteed to specification across minimum and maximum voltage levels.

Temperature also affects byte-write and block-erase performance. Low temperatures cause block-erase times to increase and byte-write times to improve (Figure 4). When cold, the breakdown voltage at the source lowers, clamping the external voltage applied for block erase. This nets a lower potential between the source and gate, slowing the tunneling process. Increased erase time results from the WSM providing extra pulses. Although electron mobility decreases at hotter temperatures, typical cells still require only one programming pulse.

Write/Erase Cycling

Intel designs extended cycling capability into its ETOX III products. For example, the 28F008SA is designed for 100,000 write/erase cycles on each of its sixteen 64-Kbyte blocks. Low electric fields, advanced low-defect oxides, and minimal oxide area per cell combine to greatly reduce oxide stress and the probability of failure.

From a performance perspective, an intrinsic mechanism occurs in long-term cycling that cause byte-write and block-erase times to increase (Figure 5), but still conform to specification. Specifically, hot electrons from programming trap in the tunnel oxide near the drain junction. This creates a negatively-charged barrier, slowing hot-electron injection. Similarly, erase times increase due to charge trapping near the source junction, making tunneling less efficient. The robustness of ETOX III minimizes these effects. Write and erase times remain consistent over the first 10,000 cycles, and typically double as the device nears 100,000 cycles.

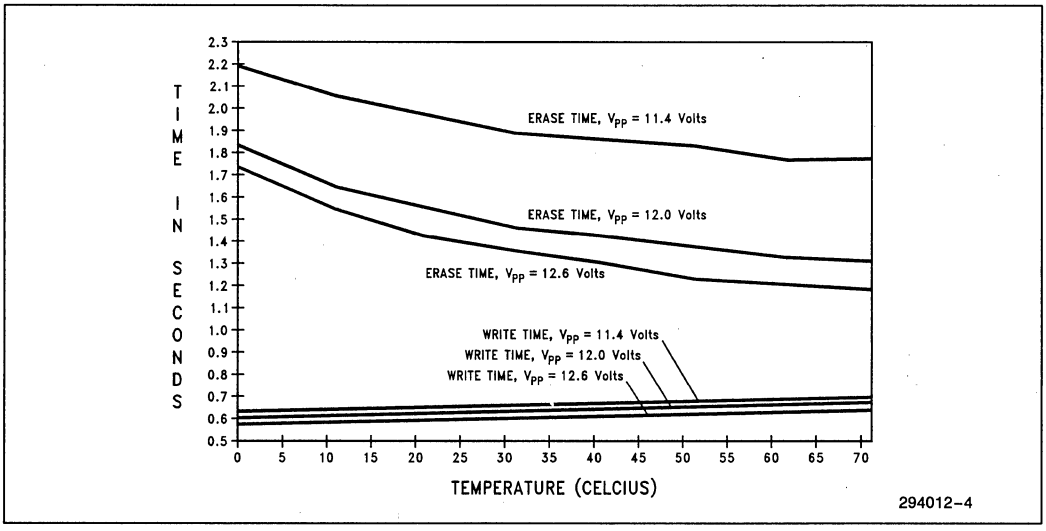


Figure 4. 28F008SA Block Write and Erase Times vs Temperature and V_{PP}

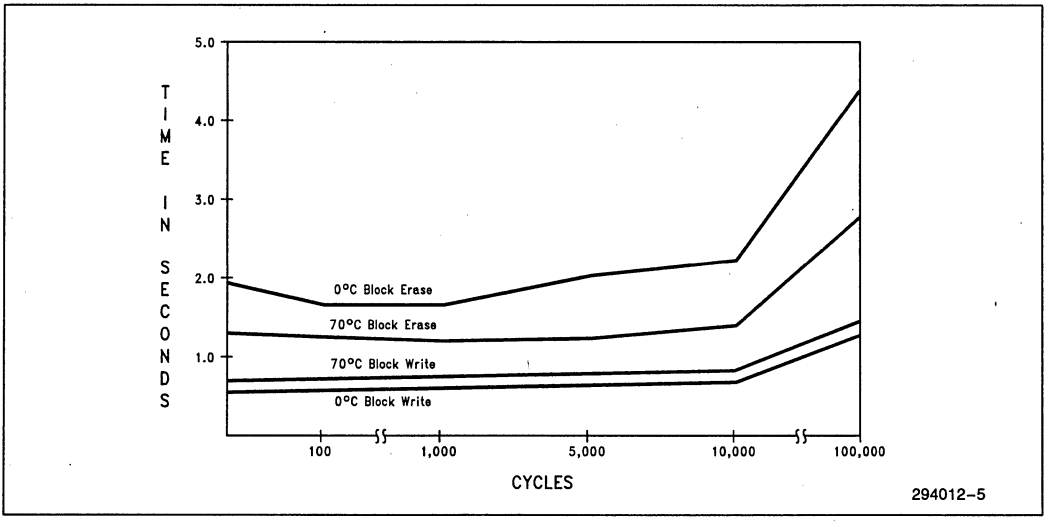


Figure 5. Write and Erase Times vs Cycling

Process Variation

Intel's process control of ETOX III critical cell dimensions keep write and erase electrical characteristics on target with little variance. This capability produces devices with consistent byte-write/block-erase times making product performance predictable.

Over the processes lifetime, internal cell dimensions may exhibit some small variance. The primary process variable affecting erase is tunnel oxide thickness. Channel length (L_{eff}) has the largest impact on programming. Outgoing product testing ensures performance to specification regardless of these minor variances.

Electrical Testing

Electrical testing provides added value to Intel Flash Memory products. This elaborate testing gives insight to device characteristics and ensures product longevity. Moreover, flash reliability qualifications assure product performance and long-term durability.

Electrical erase at wafer and package test allow high confidence of detecting oxide defects. This electrical testing also ensures that outgoing products perform to specified temperature extremes. Optimization of ETOX III process and designs, developed from previous ETOX generations and continuous data collection, yield a very manufacturable and cost-effective technology.

ETOX III FLASH MEMORY ARRAY CONSIDERATIONS

Intel Flash Memory architecture has evolved from bulk arrays (full-chip electrical erase), to array segmentation referred to as blocking. Blocking divides the device memory array into smaller sections that function as individually-erasable units.

28F008SA Array Architecture

Figure 6 illustrates the 28F008SA. Sixteen equal 64-Kbyte blocks make up this 8,388,608-bit memory array. Each block consists of 512 columns by 1024 rows. Columns in each block are further subdivided into eight input/outputs, each containing 64 columns. Typical block-erase and block-write times for this device are 1.6 and 0.6 seconds. Typical byte-write time, including WSM overhead, is 9 μs.

28F008SA Byte-Write Operation

During byte write, column address decoding determines which eight bitlines of a target block connect to the intermediate programming voltage ($\sim V_{PP}/2$). Row decoding determines wordline drive to V_{PP}. For example, writing a byte of data in block 0 sets that one wordline to V_{PP}. Address selection sets all other wordlines in the array to ground. Array decoding/layout and cell durability assure device performance and reliability, and long-term data retention.

28F008SA Block-Erase Operation

Erasing a block involves simultaneous erasure of all bits in that block. For example, erasing block 0 sets all block 0 sources to V_{PP} and all block 0 wordlines to ground. Address decoding drives all other wordlines, bitlines and sources in the array to ground. This eliminates the possibility of corruption to data stored in non-selected blocks.

28F008SA Cell Voltage Threshold

Efficient blocking layout and optimized decoding result in smaller die size. Blocking tightens program and erase V_ts by dividing process variation into smaller regions. The internal WSM algorithms and their associated program/erase and verify circuits also keep V_t variations to a minimum. This design for manufacturing approach increases product stability.

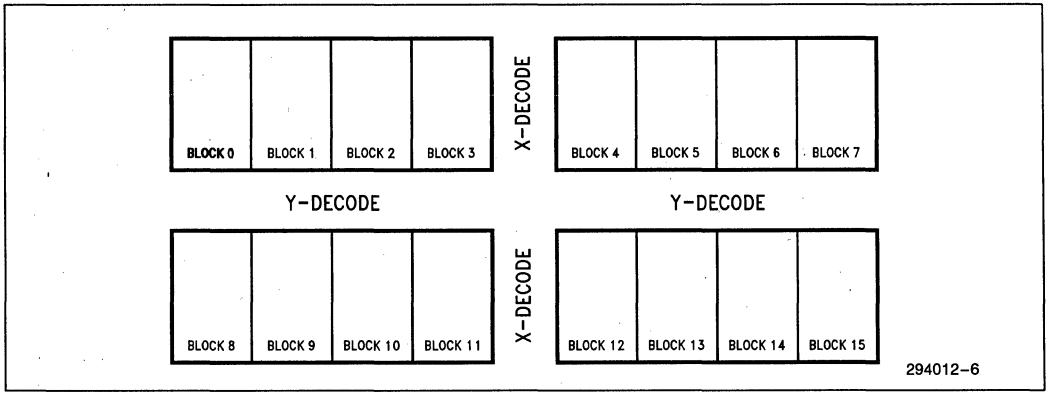


Figure 6. Intel 28F008SA Memory Array Layout

FLASH VS. OTHER SEMICONDUCTOR MEMORY TECHNOLOGIES

Intel's scaling advances in flash memory manufacturing and design provide optimal cell/array compaction. In roughly twenty years, Intel non-volatile memory density has gone from 2,048 bits to 8,388,608 bits, a 4096x increase. Figure 7 compares other memory types to show relative density progression. The fast ramp in ETOX flash memory density results from its similarity to EPROM.

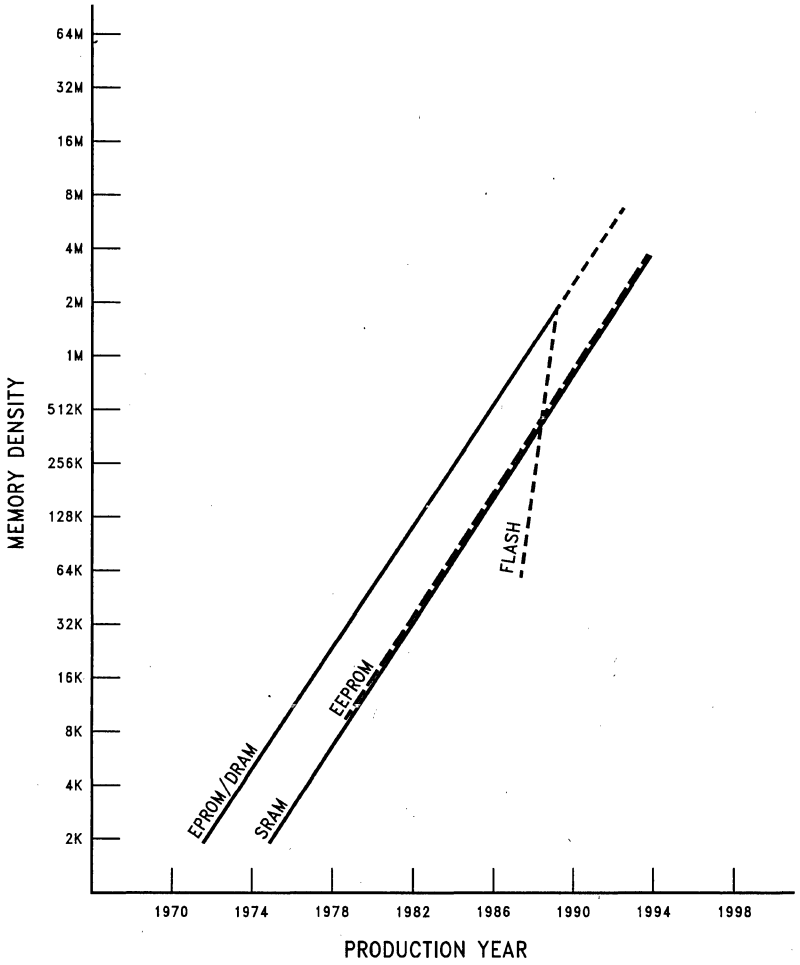


Figure 7. Memory Evolution

9

Figure 8 illustrates the relationship between cell sizes of different memory types and minimum geometries. As dimensions scale, certain memory types become cellsize limited (i.e. some components cannot shrink proportionally). The memory cost per bit learning curve shows flash in a strong position. This curve, shown in Figure 9, reflects how Intel's experience reduces cost for increased memory density.

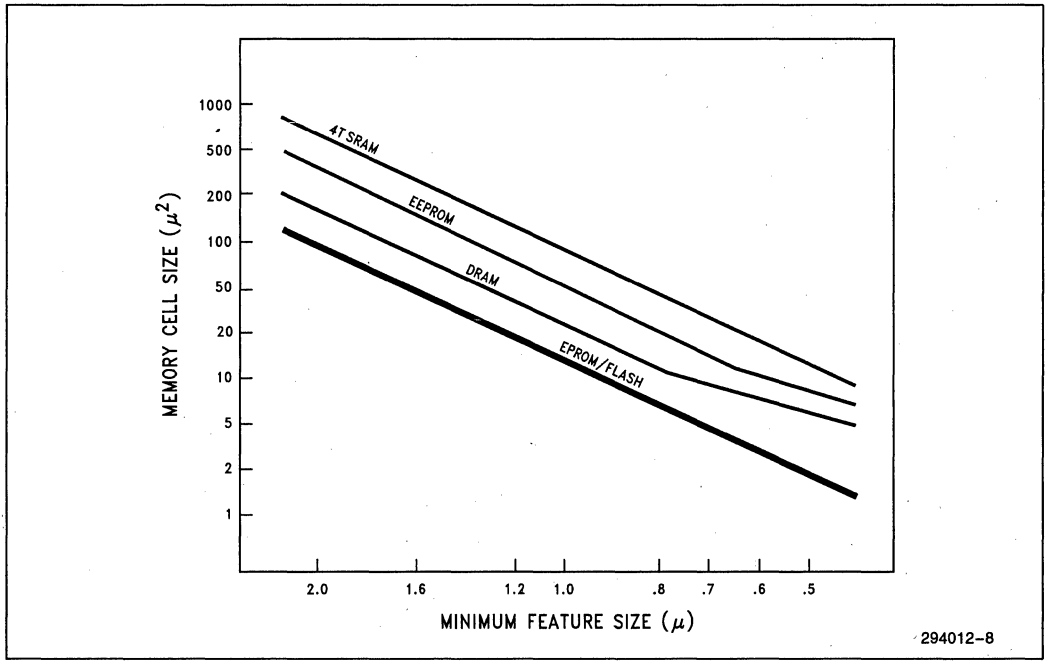


Figure 8. Memory Cell Size Trends

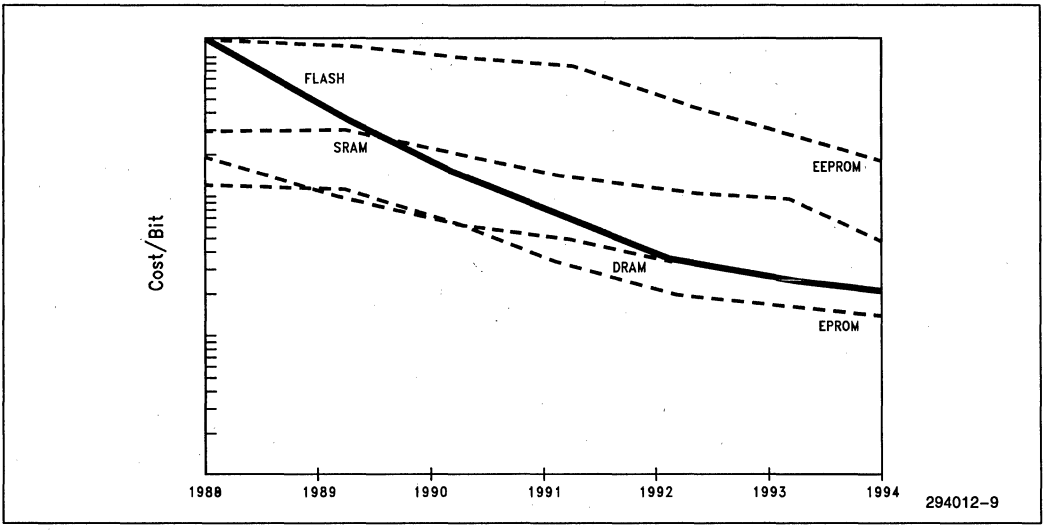


Figure 9. Memory Cost/Bit Learning Curves

Since the late 1980s, a new memory sub-system has arrived on the market, offering an alternative to high-density file system media. Intel's Series 2 Flash Memory Cards take advantage of the 28F008SA and its second-generation architecture to provide card densities of up to 20 Mbytes and new functionality. This relatively new technology offers a solid-state file system (Figure 10) that will double in density with new ETOX generations.

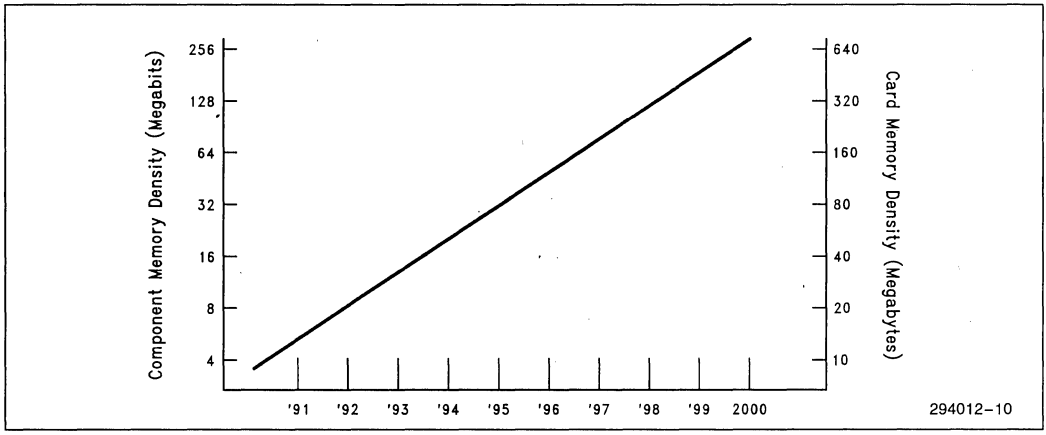


Figure 10. ETOX Component and Memory Card Density Over Time

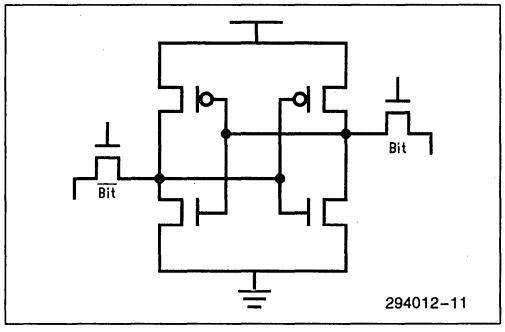


Figure 11. Six-Transistor SRAM Cell Schematic

SRAM and DRAM

SRAM and DRAM have fast read/write speeds. Both are volatile memories requiring continuous power to retain data. Standard SRAMs (Figure 11) require four to six transistors for each flip-flop cell. This greatly reduces memory capacity per unit area, raising product cost for a given density.

DRAM requires constant refresh of the capacitor-like storage mechanism (Figures 12 and 13) due to leakage currents and read operations. Charge storage requirements limit size reduction of the capacitor, which in turn limits memory array compaction. With smaller geometries, DRAM cell structures are more complex, requiring more process manufacturing steps.

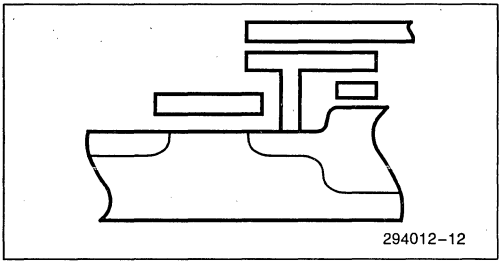


Figure 12. Stacked DRAM Cell (Side View)

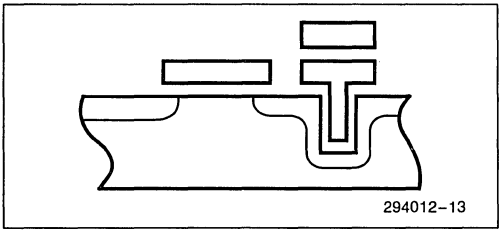


Figure 13. Trenched DRAM Cell (Side View)

9

Most DRAMs require a read parity bit for two reasons. First, alpha particle strikes can disturb cells by ionizing radiation, resulting in lost data. Second, when reading DRAM, the cell's storage mechanism capacitively shares its charge with the bitline through a select transistor. This creates a small voltage differential to be sensed during read access. This low voltage differential can also be influenced by nearby bitline voltages and device noise.

ETOX III floating-gate technology electrically isolates the substrate from the charge storage mechanism. Unlike DRAM, floating-gate charge determines cell V_ts, which in turn controls bitline voltages. This allows flash memory read sensing to easily detect cell V_ts. A current swing of 70 μA, from strong cell V_ts, make bitline voltage transitions a key factor to fast read speeds.

Most DRAMs require high active power consumption. Charge storage requirements and read signal strength constrain DRAM cell compaction. Low-power ETOX III flash memory has a simple single-transistor cell with only minor scaling limitations through the year 2000. This results in a mainstream memory that does not need power to retain data.

EPROM

Intel's EPROM and Flash Memory cells share a common stacked-gate profile (Figure 14), with two basic differences relating to their respective erase mechanisms. EPROM requires ultraviolet light to erase; flash erases electrically. For this reason, flash has a thinner cell oxide to allow Fowler-Nordheim tunneling, and a deeper source junction to prevent breakdown during erase.

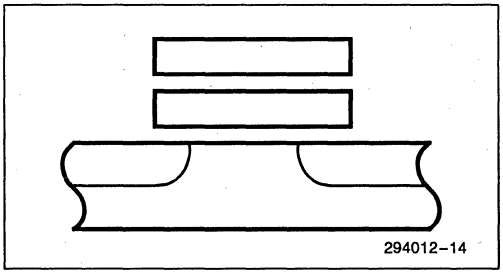


Figure 14. EPROM Cell (Side View)

EPROM technologies that migrate toward smaller geometries make floating-gate discharge (erase) via UV exposure increasingly difficult. One problem is that the width of metal bitlines cannot reduce proportionally with advancing process technologies. EPROM metal width requirements limit bitline spacing, thus reducing the amount of high-energy photons that reach charged cells. EPROM products built on submicron technologies will face longer UV exposure times.

Intel's ETOX III technology employs double-metal processing to strap wordlines in metal for improved read performance. This advance is not likely to appear on EPROM because it would block even more UV light. Since flash memory electrically erases, it eliminates these concerns. Moreover, flash electrical erasure eliminates the UV window and its associated cost, and allows for the most advanced and innovative plastic surface-mount packaging solutions.

EEPROM

Conventional two-transistor EEPROM cells (Figures 15 and 16) limit layout density. This is primarily due to the second transistor (bit select) and associated decoding required for single byte program and erase capability. Technology design requirements make EEPROM cells, like triple poly, significantly larger than flash. Typical EEPROM technologies are more complex, making wafer manufacturing difficult and expensive.

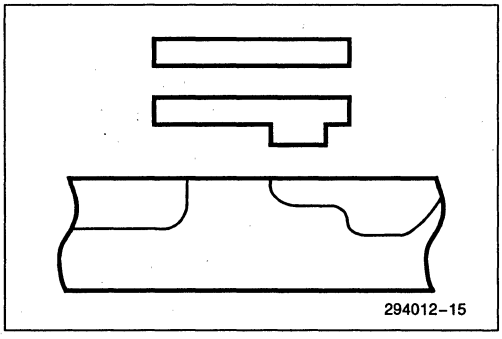


Figure 15. Flotox EEPROM Cell (Side View)

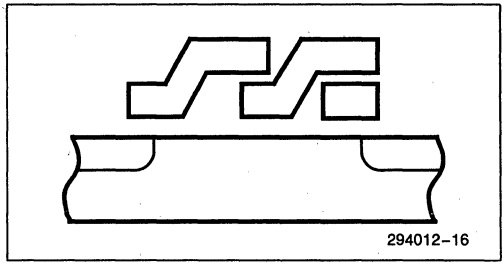


Figure 16. Triple Poly EEPROM Cell (Side View)

Because of their traditional application, EEPROMs use a very high internal voltage (17V to 30V) to achieve fast program and erase times. These high voltages and resulting electric fields cause cell oxides to breakdown, shortening cycling life and degrading cell thresholds. Additionally, this high voltage stresses periphery transistors. Intel's Flash Memories are more akin to EPROM; both use a significantly lower voltage around 12V.

NAND

Futuristic types of EEPROM (Figure 17) that have shifted from highly-manufacturable NOR-gate architectures, appear to provide ETOX-like density on a per cell basis and potential use in similar applications. A closer examination reveals internal positive and negative charge pumping. When applied across a memory cell, the dual charge pumps net a high voltage that cause oxide stress. These stacked-gate cells program and erase via tunneling. They program from the substrate to the floating gate, and erase in the reverse bias.

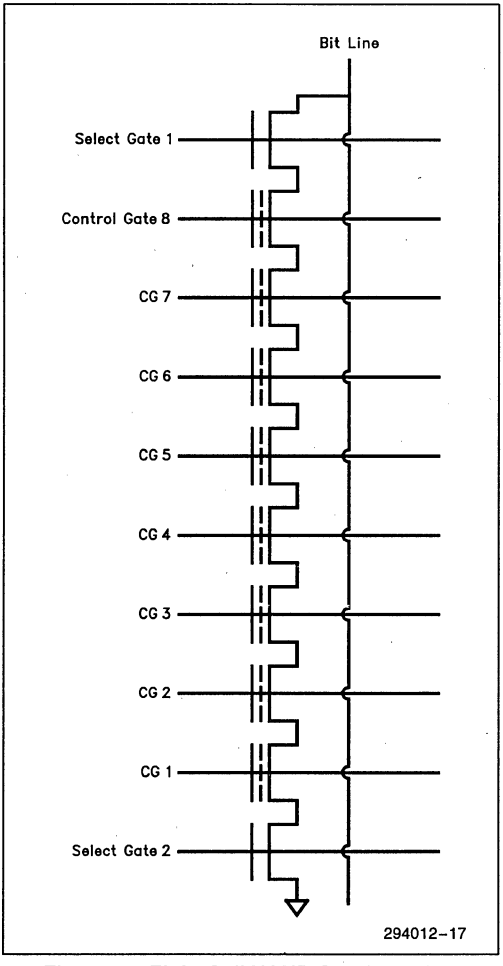


Figure 17. Eight Cell NAND Configuration

Layout of the NAND array groups eight cells as a set, each set requires two select transistors to control bitline access. Due to this NAND configuration, the read path goes through other cells making access slow. Increased read speed requires an internal SRAM page buffer, however the first read always remains slow. Fast write times require a page buffer for full wordline programming. Several wordlines erase at once, setting the block size. Product scaling becomes limited from high-voltage requirements on periphery transistors and isolation technology. Compared to EEPROM, the NAND approach improves array compaction at the expense of more complex decoding and periphery circuitry.

SUMMARY

Intel's technology advances result in products that are more efficient, more reliable, less expensive and higher performance. Submicron geometries and double-metal technology allow considerable memory array compaction, providing increased read and byte-write performance. Design compaction also improves with EPI wafers that reduce latch-up, allowing closer transistor layout. Strong cycling endurance results from the quality of the thin low-defect tunnel oxide, and the electrical characteristics of internal program and erase operations. Cycling, voltage and temperature exhibit only a small influence on byte-write and block-erase speeds. Products built on Intel's CMOS ETOX III flash memory technology require minimal power consumption during writes, erasure, read, and low-power sleep or standby modes.

Intel Flash Memory products designed on ETOX III will satisfy many different applications.The Thin Small-Outline Package (TSOP) provides customers with high memory density in the smallest footprint. Some applications for ETOX III flash products include memory cards, solid-state drives, non-volatile operating systems, high-performance system storage, data acquisition, and application and embedded code storage. The solid-state nature of flash results in improved ruggedness over mechanical rotating media. With blocking, applications can perform background erase to optimize system performance. Today, Intel's technological advances in flash memory are driving cost to parity with DRAM. This steep decline in the price learning curve enables new classes of systems and system architectures.

OTHER REFERENCES

Related documents of interest to readers of this engineering report:

28F008SA Data Sheet (Order No. 290429)

Series 2 Memory Card Data Sheet (Order No. 290434)

ER-27: The Intel 28F008SA Flash Memory (Order No. 294011)

APPENDIX A

First-generation flash devices, like the 28F020, use externally-controlled algorithms for byte write and bulk erase. These algorithms require that customer software control:

- Cell pulse widths, and pulse repetition where required.
- Erase preconditioning (i.e. pre-programming all cells before erase).
- Timeout delays to allow cell voltages to transition from program or erase levels to read verify levels.
- Read compare operations to determine success.

Second-generation architectures, like that on the 28F008SA, contain an internal Write State Machine (WSM) to simplify software development. This WSM provides internal control of all of these first-generation requirements, as well as reporting on activity progress/success through the internal Status Register. A dedicated output on the 28F008SA allows immediate hardware signaling of WSM activity status. The Command User Interface (CUI) provides customer control.

The other major architectural feature of second-generation devices is array segmentation, also referred to as "blocking". First-generation products erase in bulk. This means that all cells in the array erase simultaneously. Second-generation "sectored" architectures divide the array into separately-erasable block segments. This provides logical segmentation of customer code, and allows reads of other device blocks while one is erasing (i.e. via the erase-suspend/resume commands).

intel®

ER-33

ENGINEERING REPORT

ETOX IV Flash Memory Technology: Insight to Intel's Fourth-Generation Process Innovation

ALAN K. BUCHECKER
ALEXANDER C. MITCHELL, III
MEMORY COMPONENTS DIVISION
INTEL CORPORATION

9

November 1994

ADVANCE INFORMATION
Order Number: 294016-002

ETOX IV FLASH MEMORY TECHNOLOGY: INSIGHT TO Intel's FOURTH GENERATION PROCESS INNOVATION

CONTENTS

PAGE

INTRODUCTION 9-21

ETOX IV FLASH MEMORY CELL 9-21

Cell Processing 9-22

Data Write 9-23

Erase 9-26

ETOX IV PROCESS CHARACTERISTICS 9-29

Write/Erase Performance with Voltage and Temperature 9-29

Write/Erase Cycling 9-29

Process Variation 9-32

Electrical Testing 9-33

ETOX IV PROCESS-RELATED IMPROVEMENTS 9-34

History and Evolution 9-34

Cell/Array Compaction 9-36

Enhanced Contact Technology 9-36

Improved Cell/Array Uniformity 9-36

Write/Erase Performance 9-36

Cycling Performance 9-37

Read Access 9-37

CONTENTS

PAGE

FLASH vs OTHER SEMICONDUCTOR MEMORY TECHNOLOGIES 9-37

SRAM and DRAM 9-40

EPROM 9-41

EEPROM 9-42

PROCESS/DESIGN TRADEOFFS TO ACHIEVE THE PERFECT FLASH MEMORY 9-42

NOR Flash 9-42

ETOX Flash 9-43

Negative Gate-Bias Erase (NGE) 9-43

NAND Flash 9-43

Triple-Poly Flash 9-45

SUMMARY 9-46

OTHER REFERENCES 9-46

INTRODUCTION

Intel's ETOX IV (EPROM with Tunnel Oxide) NOR-based Flash Memory technology builds on three previous generations of ETOX process learning, while also leveraging decades of stacked-gate memory cell manufacturing experience dating back to the development of the first EPROM in 1971. This fourth-generation ETOX process yields devices based on 0.6 μm minimum-design-rule geometries.

Intel's ETOX Flash Memories combine state-of-the-art EPROM programming methodology with EEPROM-like in-system electrical erasure. Sectoring of the large memory array into smaller independent-erase blocks, coupled with on-chip automation of their write and erase algorithms, enables a flash product that can be termed a Block-Erasable Non-volatile Random-Access Memory. This functionality, combined with Intel's accumulated years of manufacturing experience, electrical-test and temperature-stress data collection, and advanced design/architectural technology innovation, yields a very manufacturable and versatile non-volatile memory that is highly reliable, extremely stable and cost effective.

Intel considered all aspects of product technology and usage, and balanced the associated tradeoffs of cell size, array architecture and system design to generate a very robust and well-rounded flash memory approach. ETOX IV cell integrity produces devices with 100,000-cycle minimum write/erase endurance per block (1,000,000 cycles are obtainable with proper software control techniques). Write/erase times and die size decrease via ETOX IV's finer photolithography, allowing Intel Flash Memories to gain on DRAM in-system performance and cost-per-megabyte advances. Manufacture on 8″ wafer technology significantly increases the number of die per wafer vs 6″ fabrication. This capacity demands investment in new capital equipment, but in the end bears yet another path to reduce product cost.

This report describes the fundamentals of Intel's ETOX IV Flash Memory cell. It provides insight to device operating characteristics and reliability with focus on ETOX IV-related performance enhancements, and compares other semiconductor and flash memory technologies. The 28F016SA 16-Mbit FlashFile™ Memory is Intel's second ETOX IV product offering. This report references the 28F016SA to explain device-level concepts and ends by highlighting important flash memory application trends.

More information on the 28F016SA can be found in the following documents (see Other References section for order numbers:

- Data Sheet: 28F016SA 16-Mbit (1 Mbit x 16, 2 Mbit x 8) FlashFile™ Memory
- Data Sheet: 28F032SA 32-Mbit (2 Mbit x 16, 4 Mbit x 8) FlashFile™ Memory
- 28F016SA 16-Mbit FlashFile™ Memory User's Manual
- AP-362: Implementing Mobile PC Designs Using High Density FlashFile™ Components
- AP-375: Upgrade Considerations from the 28F008SA to the 28F016SA
- AP-377: 28F016SA Software Drivers
- AP-378: System Optimization using the Enhanced Features of the 28F016SA

ETOX IV FLASH MEMORY CELL

ETOX IV is a 0.6 μm, double-metal, double-polysilicon, N-well and P-well CMOS manufacturing process. Its lithographic advance improves memory cell and array compaction more than two-fold over its predecessor, the 0.8 μm ETOX III process.

Such a significant level of compaction greatly lessens the economic impact associated to incorporate additional logic. Consequently, advanced circuits can be cost-effectively implemented to provide the system designer with enhanced product flexibility and operating characteristics, and new modes of user functionality. Additionally, ETOX IV compaction allows for a 4x density growth path given current packaging constraints (i.e., the leadframe of the 14mm-long x 20mm-wide x 1.0mm-thick 56-ld Thin Small-Outline Package).

Data-Write performance dramatically improves via 0.6 μm geometric scaling advances. Raw cell programming time decreases to 2.5 μs, nearly halving the 4 μs pulsewidth of ETOX III products, and a 4x reduction from 10 μs on 1.0 μm ETOX II products.

Erase performance also improves. 28F016SA Block Erase (64 KB block size) is typically 800 ms. This duration is 50% faster than the ETOX III 28F008SA, with identical block size. From the process perspective, erase timing gains come from the aforementioned boost to cell programming performance (during Block-Erase pre-conditioning) and a higher degree of cell and array uniformity. Streamlining and enhancement of the internal state machine's algorithm and support circuitry contribute design improvements that speed up erase.

Cell Processing

Intel's single-transistor 0.6 μm ETOX IV Flash Memory cell, with dimensions of 1.8 μm x 2.0 μm, measures a mere 3.6 μm^2. Figure 1 shows a photomicrograph cross section of this cell.

A floating polysilicon gate (poly-1) provides the charge storage mechanism used on both ETOX flash and EPROM technologies. This component is depicted at the center of the photomicrograph. Intel's ultra-pure thin-oxide formation technology produces a low-defect tunnel oxide (~ 100Å) which separates the floating gate from its silicon interface.

Cell construction locates a poly-2 control gate directly above the floating gate. Inter-poly dielectrics isolate the floating gate from the control gate. Both the floating and control gates have additional isolation between them and their respective source/drain regions (side or spacer oxide). Tungsten silicide, deposited on the control gate, reduces wordline resistance. A dielectric layer isolates metal-1 from the control gate.

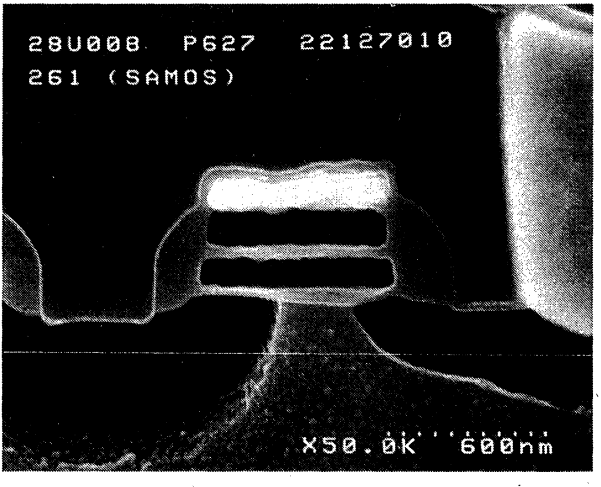
A graded source diffusion improves erase efficiency. This grading is achieved via a phosphorous implant incorporated into the source junction. The deeper implant also improves reliability by preventing source/oxide junction breakdown during erase.

As with ETOX I, II and III, metal-1 carries bitline data to the sense amplifiers and routes voltages to cell sources. Throughout the array, metal-2, which was first used on ETOX III, straps the wordline to reduce its resistance. This technique enhances device read (t_{ACC}) performance and the routing of signals and voltages during program and erase operations.

Double-metal technology also aids t_{ACC} improvements via its role in compacting the array. EPI wafer processing, which practically eliminates latch-up, also helps read-access timings by allowing tighter transistor layout.

In addition to smaller geometries, ETOX IV offers several other major advancements to this process generation via new manufacturing and contacting capabilities. These advancements tighten cell dimensions and improve step coverage.

New processing steps maintain very strict manufacturing adherence to tight cell dimensional tolerances. Cumulatively, these individual adherences translate into a more uniform array. Tighter dimensioning at the cell level enhances individual source-to-floating-gate coupling consistency, and therefore erase efficiency. Spread across the full array, this provides greater control of erase-threshold distribution.



294016-1

Figure 1. Photomicrograph of ETOX IV Flash Memory Cell (Side View)

New technologies and materials improve mask layer interconnecting through the narrower contact holes of ETOX IV geometries. This enhanced step coverage reduces fallout at wafer electrical test.

Tighter cell dimensions, improved array uniformity, and superior step coverage in both the array and periphery also contribute toward a higher degree of cell/array compaction and a more planar silicon formation. In combination with erase threshold control improvements and reduced electrical test fallout, the following device characteristics improve:

- Yields (and thus cost)
- Erase Performance
- Moisture Performance
- Reliability
- Durability
- Stability

Data Write

Writing data to an addressed memory location transitions selected cells from the "1" (erased or no charge) state to the "0" (charged) state. This involves the EPROM programming mechanism of Channel Hot-Electron (CHE) injection. When programming (Figure 2), a cell's control gate links to the external V_{PP} supply voltage via wordline connection and row decoding. The drain sees an intermediate level of $\sim V_{PP}/2$ via bitline connection and column decoding; the source is at ground.

V_{PP} on the control gate (V_{CG} = 12V) capacitively couples to the floating gate through the intervening dielectric. This coupling raises the floating gate to a programming bias voltage, which in turn inverts the p-type channel underneath. After channel inversion (now taking on characteristics of n-type material), the aforementioned intermediate-level voltage is applied to the drain. This drain voltage attracts electrons, the majority charge carrier in n-type material, from the source.

As the drain voltage ramps, channel electrons gain a higher drift velocity with resulting increased kinetic energy. Collisions between them and substrate atoms energize the silicon lattice. At that point, these channel electrons cannot transfer their kinetic energy to the atoms fast enough to maintain a thermal balance. They become "hotter" and many scatter toward the tunnel oxide. At some point they overcome the 3.1 eV (electron voltage) tunnel-oxide barrier and accumulate onto the floating gate.

Electron storage on the floating gate creates a negative potential opposing voltage (e.g., read or margin-verify bias) applied to the control gate. This opposition results in a higher turn-on threshold voltage (V_t, Figure 3) for that transistor (memory cell). An unprogrammed cell has a turn-on $V_t < V_{te}$; programming increases this threshold to $> V_{tp}$. This threshold is guaranteed by an internally-conducted margin-bias read producing a current (I_{PMrgn}) that is sensed against a program reference circuit providing a current relationship to V_{tp} (I_{PRef}). See Figures 4 and 5 for detail of the program reference circuit and current.

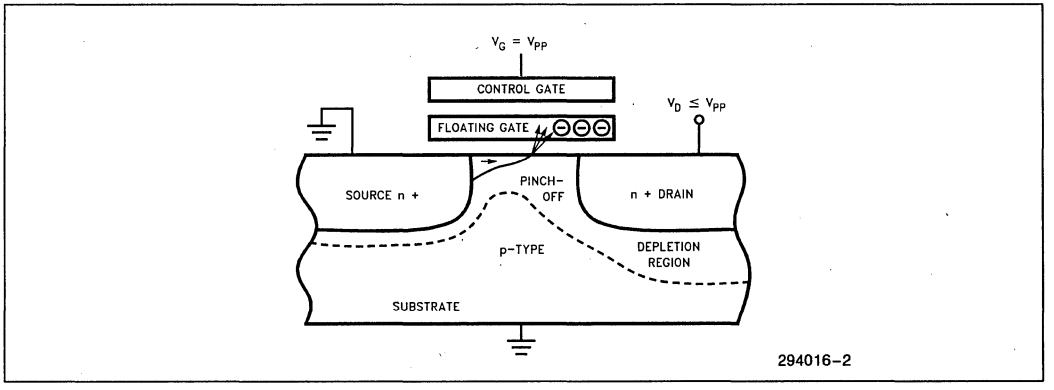


Figure 2. ETOX IV Flash Memory Cell during Programming (Side View). CHE Injection produces electron migration across the tunnel-oxide barrier; the electrons store on the isolated floating gate, raising this cell's turn-on threshold.

9

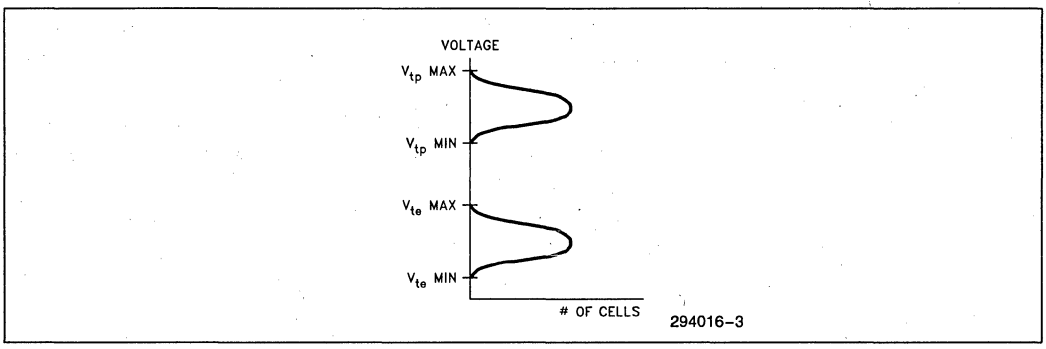


Figure 3. ETOX IV Program and Erase Thresholds. The upper plot shows V_t distribution for "0" cells (programmed), the lower plot displays for "1" cells (erased).

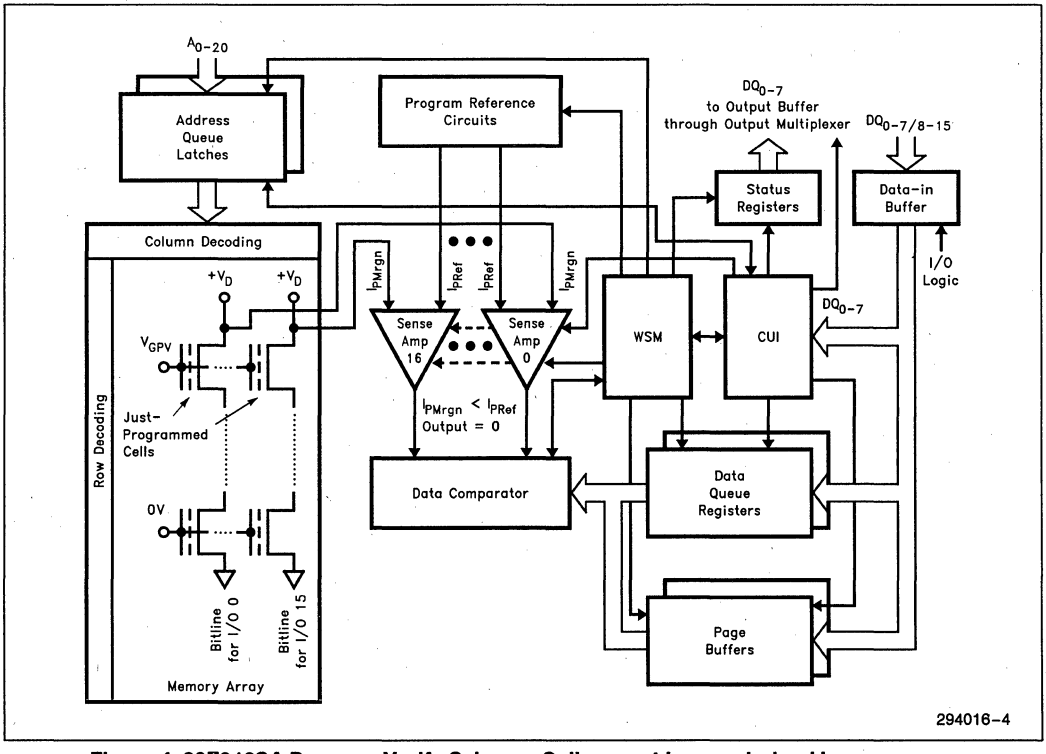


Figure 4. 28F016SA Program Verify Scheme. Cell current I_{PMrgn}, derived by a program margin bias, is sensed against current I_{PRef} (from Program Reference Circuit). The sensed outputs are then compared to the original data written and stored in a Data Queue Register.

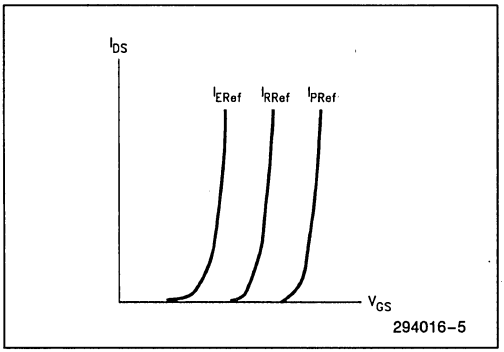


Figure 5. ETOX IV Program, Erase and Read Reference Currents. The reference circuitry, shown in Figures 4, 6, 8 and 9, provides these currents for sensing against cell program and erase verification currents, and read currents during memory access.

During read operations (Figure 6) V_{CC} (or a boosted voltage for 3.3V operation) is placed on the control gate, the source is grounded and a drain bias is applied. In this mode, a programmed cell conducts little to no current. The bitline (column) current for this cell is sensed against the current produced by a read reference circuit. If the current in the read reference circuit is greater than that through the programmed cell, a "0" results at the output of the sense amplifier.

The internal ETOX IV programming pulse is 2.5 μs; this optimized pulsewidth yields faster Data-Write performance and greater cycling reliability. The 28F016SA's on-chip write/erase automation circuits, governed by a functional unit called the Write State Machine (WSM), impart an additional 3.5 μs of overhead to the programming operation. The combined 6 μs duration begins with the last WE# rising edge (WE#-controlled timings) of the command/data bus-cycle string for a Data-Write sequence. This 3.5 μs overhead is required to activate WSM support circuits, decode programming voltages onto correct cell locations and run through the flow of the internal algorithm.

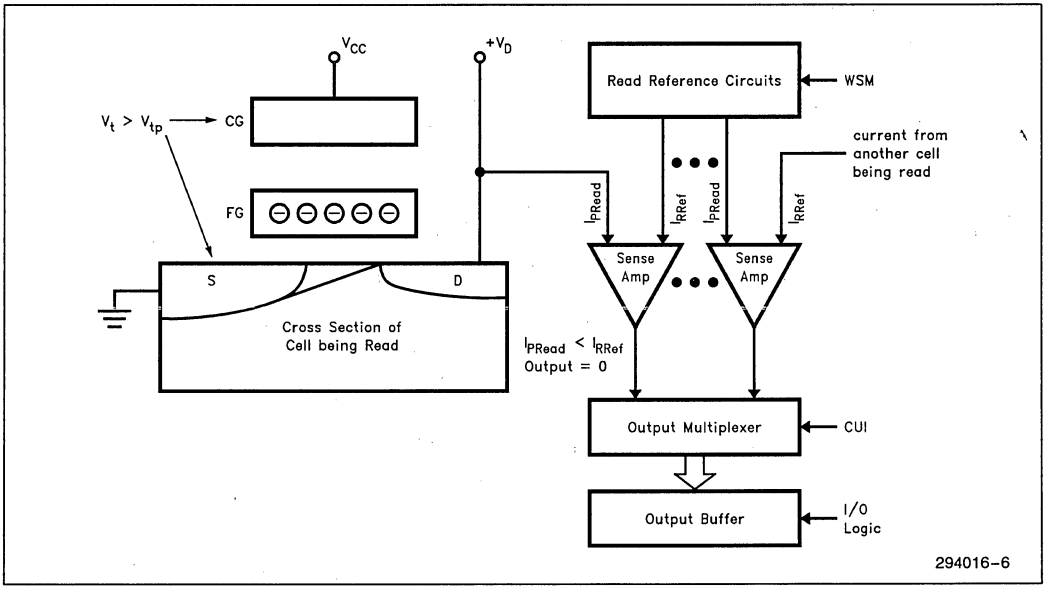


Figure 6. ETOX IV Flash Memory Read Scheme (Programmed Cell). The negative charge stored on the poly-1 floating gate (FG) increases this cell's turn-on threshold, thus with V_{CC} applied to the control gate (CG) its current $I_{PRead} < I_{RRef}$ (Read Reference Current).

9

Additionally, part of this same overhead is needed for voltages to switch from a programming bias to an elevated read bias in preparation for margin sensing against a program reference current. This margin-sense activity is followed with a comparison to the original data written and held by an input register (see Figure 4 of this report, and the sensing and data comparator sections of the 28F016SA Block Diagram in Data Sheet or ER-31).

Cell current I_{PMrgn}, which in the case of Data-Write verification is derived from an elevated read bias on that cell, is fed, as well as program reference current I_{PRef}, into a sense amplifier. The output of multiple sense amplifiers (8 for byte-wide operations, or 16 for word-wide operations) is routed into a data comparator for collation with the original input held by one of the data-in queue registers. The comparator signals the WSM with the outcome of this data-compare operation. Like previous Intel Flash Memory products, the automated WSM allows for the occasional word or byte requiring more than one pulse.

When programming completes, the WSM updates the Ready/Busy# (RY/BY#) bit of the device's internal Status Registers and the dedicated RY/BY# acknowledgement pin. Additional status bits detail success/error information, and the probable reason for failure, if one occurs.

Erase

Erasure forces all memory cells in selected blocks to the "1" (no charge) state. This action involves an EEPROM-like erase mechanism called Fowler-Nordheim (F-N) tunneling. During erase, V_{PP} is routed to all source connections within a single block via the appropriate block source switch (see Figure 10, and its corresponding text in ER-31). All control gates are grounded and the drains for the block to erase are floated (see Figure 7 of this report for individual cell configuration); all other block drains are grounded.

While a cell is biased in this fashion, electrons previously stored on the floating gate now attract towards the source. The electrons tunnel through the thin oxide layer, producing the desired effect of lowering that cell's V_t. This V_t is verified $< V_{te}$ in a method similar to programming operations (see Figure 8 for Erase-Verify diagram); the difference being, erase reference (Figure 5) is adjusted to a current relationship with V_{te}.

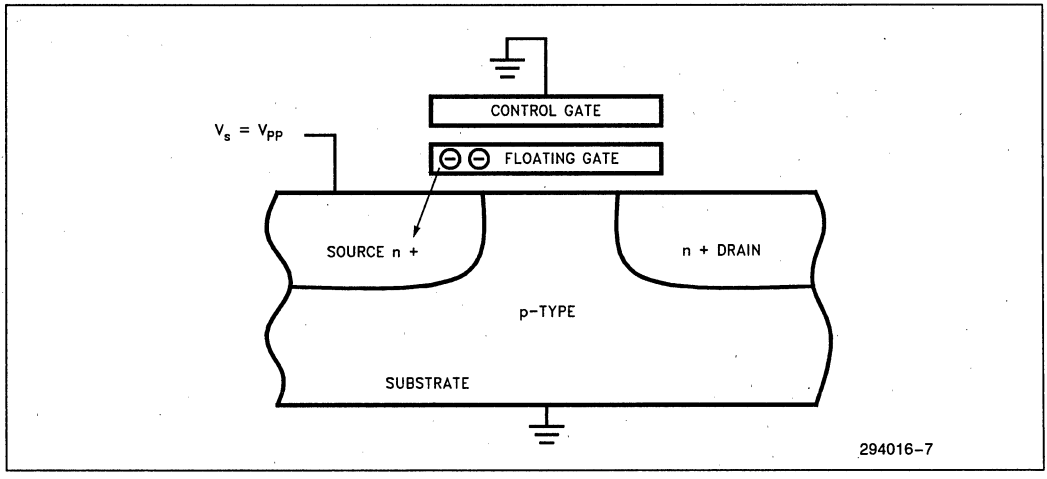


Figure 7. ETOX IV Flash Memory Cell during Erase (Side View). F-N tunneling frees electrons so that they can migrate from floating gate to source. This action returns the cell V_t level to the unprogrammed state.

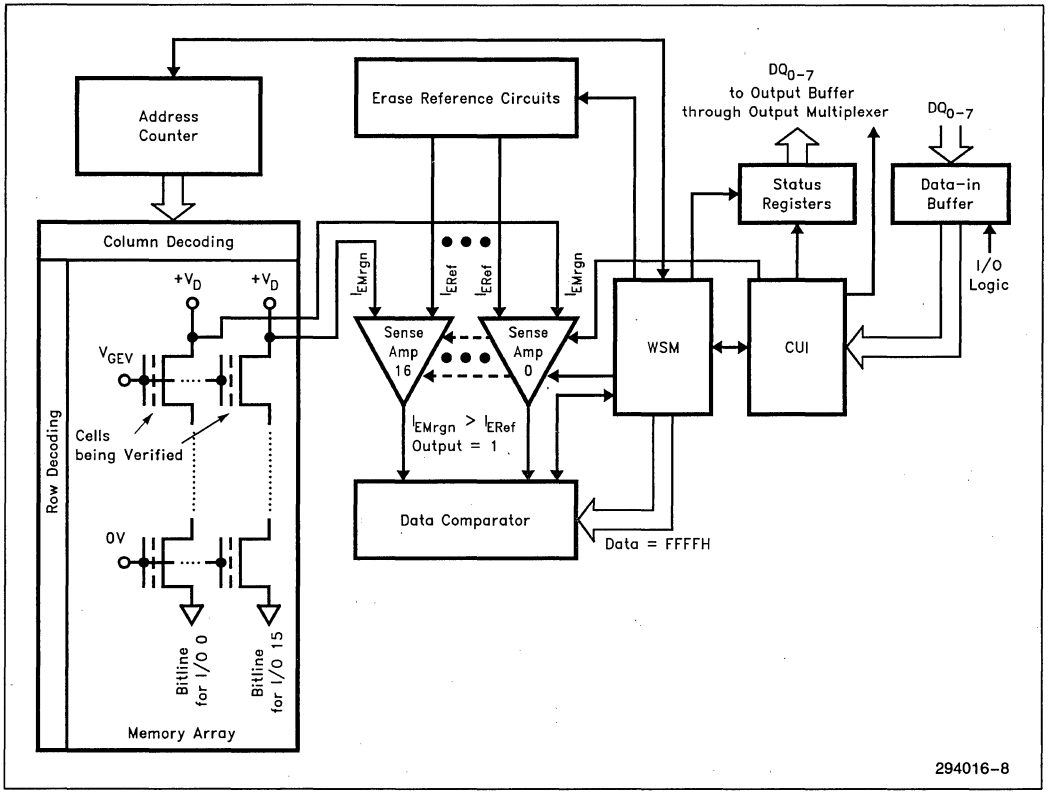


Figure 8. 28F016SA Erase Verify Scheme. Cell current I_{EMrgn}, derived from a erase-margin bias, is sensed against current I_{ERef} (from Erase Reference Circuit). The sensed outputs are then compared to data FFFFH.

Without a floating-gate-stored negative potential to overcome, erased-cell thresholds produce a higher current relative to the read reference circuit. During read operations then, a "1" is seen at the output (Figure 9).

Since NOR-type flash memories do not contain the two-transistor storage-and-selection cell structure of EEPROM technology, they cannot erase with word/byte granularity. Instead, flash memories erase in bulk (entire device) or via blocks (sectored portion of array). This means that many cells have their floating-gate charge depleted simultaneously.

To avoid device damage (i.e., over-erasure), the internal flash-erase algorithm (or the external customer-controlled algorithm on older products) first pre-conditions the block to be erased (i.e., programs all cells to "0"). After successfully completing this portion of the erase operation, the WSM can then initiate erase pulsing.

The ETOX IV erase pulse is 10 ms; this pulse duration is optimized and balanced for fast Block-Erase time and enduring cyclability. After each pulse attempt, the WSM signals for margined sensing against the erase reference current, and compare to data value FFFFH for all word addresses within the associated block.

9

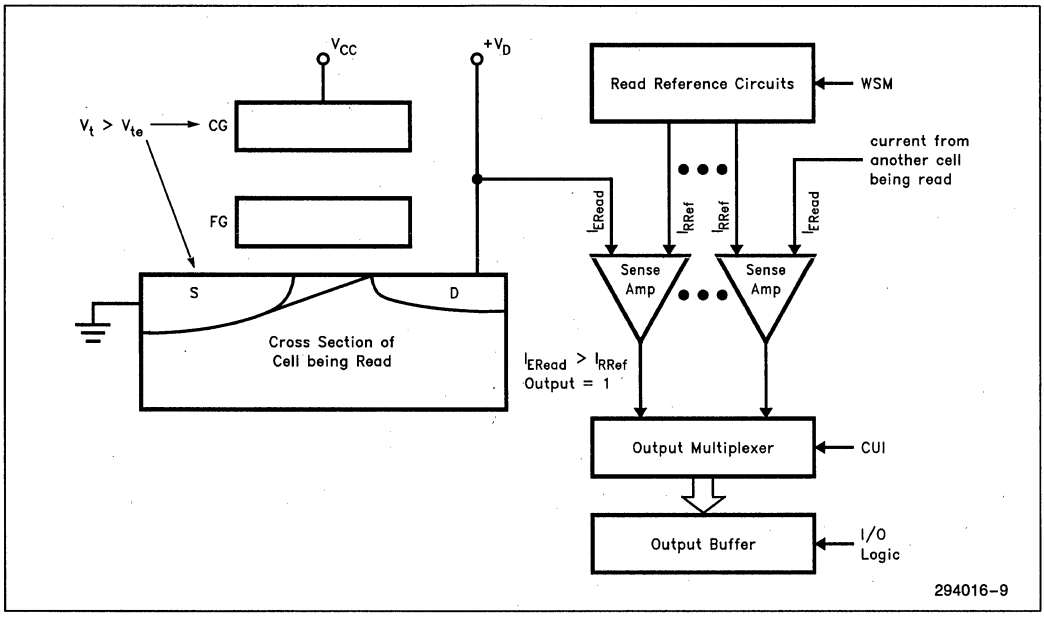


Figure 9. ETOX IV Flash Memory Read Scheme (Erased Cell). The absence of floating-gate (FG) charge gives this cell a turn-on threshold similar to a MOSFET. Therefore, with V_{CC} applied to the control gate (CG), its current $I_{ERead} > I_{RRef}$ (Read Reference Current).

An internal address counter controls selection for the array data location to be sensed (see Figure 8 of this report, and the address counter section of the 28F016SA Block Diagram in Data Sheet or ER-31). This counter starts at the last word location within a block, then decrements its pointer through the remaining addresses via increments of one.

The WSM halts this internal margin-sense/data-compare function when a word location does not verify. In this instance, another 10 ms erase pulse is applied. The location last checked is then re-tested, and if successful, address sequencing and verify-testing recommences; if not, another erase pulse is applied. This testing, re-pulsing and re-testing flow continues until all block memory words confirm erased.

As with programming, the WSM updates Status Register RY/BY# and dedicated hardware RY/BY# indicators at erase completion. Additional register bits provide success/error status, and the probable reason for failure, if one occurs.

ETOX IV PROCESS CHARACTERISTICS

This section expands on key device performance, stability and reliability characteristics relative to the process, showing influences of voltage and temperature. The following topics are covered:

- Write/Erase Performance with Voltage and Temperature
- Write/Erase Cycling
- Process Variation
- Electrical Testing

Write/Erase Performance with Voltage and Temperature

Voltage influences Data-Write and Block-Erase performance. Maximum V_{PP} bias significantly reduces Block-Erase times, while providing only slight improvement to Data Write (i.e., averaged Data-Write times over numerous operations). The effect on erase is more pronounced because Data-Write operations typically require a single programming pulse at nominal V_{PP}, therefore gains from raising V_{PP} are not discernible on single Word/Byte Writes. Conversely, reduction in the number of pulses required for Block Erase, when using V_{PPMAX}, is perceptible.

V_{CC} also influences Data-Write and Block-Erase times. The WSM is designed using clocked logic circuits, with a ring oscillator generating their clock signals. The frequency of a standard ring oscillator varies with processing, temperature and supply voltage. Improved designs minimize these variations, but some level will always exist. For Data Write, V_{CC}'s influence on the oscillator's operating frequency imparts the primary effect to these operations (as mentioned above, V_{PP}'s effects are secondary). V_{CC} min-max variances, while in 5.0V or 3.3V mode are negligible, but the difference between modes is observable. For Block Erase, V_{CC}'s influence is secondary, about $\frac{1}{2}$ to $\frac{1}{3}$ the impact seen by varying V_{PP} to minimum, nominal and maximum levels. Despite these dependencies, write and erase times are guaranteed to specification across all minimum and maximum voltage corners.

Temperature also affects Data-Write and Block-Erase times. Low temperatures cause a decline in Block-Erase performance, but Data-Write operations improve. When cold, the breakdown voltage at the source lowers, thus clamping the internal Block-Erase voltage. This nets a lower potential between source and gate, slowing the tunneling process. The time increase results from extra WSM pulses.

Cold temperatures enhance Data-Write performance because electron mobility improves. Increased mobility yields a higher rate of electron travel toward the drain producing fewer, but stronger collisions with substrate atoms. These collisions, now more fierce, impart greater kinetic energy to the electrons being freed. This energy gain aids their migration across the isolation barrier of the tunnel-oxide layer, resulting in greater electron accumulation onto the floating gate per unit time. Although electron mobility decreases at hotter temperatures, typical cells still require only one programming pulse.

Figures 10 and 11 depict the voltage and temperature dependencies of 28F0I6SA Word-Write and Block-Erase times. For Figure 10, it is shown that higher V_{CC} levels and colder temperatures improve Data-Write performance. Figure 11 reveals maximum V_{PP} and hot temperature enhances Block-Erase times, as well as high V_{CC} bias.

Write/Erase Cycling

Cycling is a critical consideration for applications like filing systems using flash-based memory cards or solid-state drives in harsh mobile or industrial computing environments, or a notebook/pad computer DRAM suspend storage array built with flash memory chips on the motherboard. Other parameters, such as read-access timings, write/erase performance, power consumption and data retention follow in order of precedence, and of course, flash solid-state ruggedness is far superior to electromechanical rotating media, hence not a concern here.

Intel designs extended cycling capability into its ETOX IV products. As an example, the 28F016SA is specified for a minimum of 100,000 write/erase cycles on each of its thirty-two 64-KB (32-KW) blocks. Furthermore, the 28F016SA can deliver 1,000,000 Block-Erase cycles provided block-retirement methods are employed. Spare blocks must be included in system designs to successfully implement block retirement methodologies.

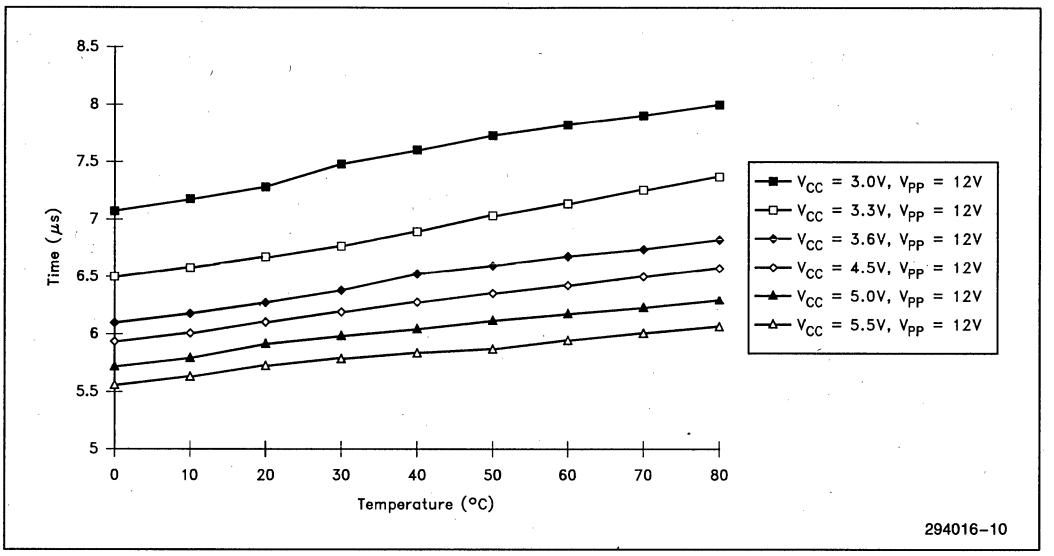


Figure 10. 28F016SA Word-Write Time vs Temperature and V_{CC}. Word-write efficiency improves at cold temperature and maximum V_{CC}.

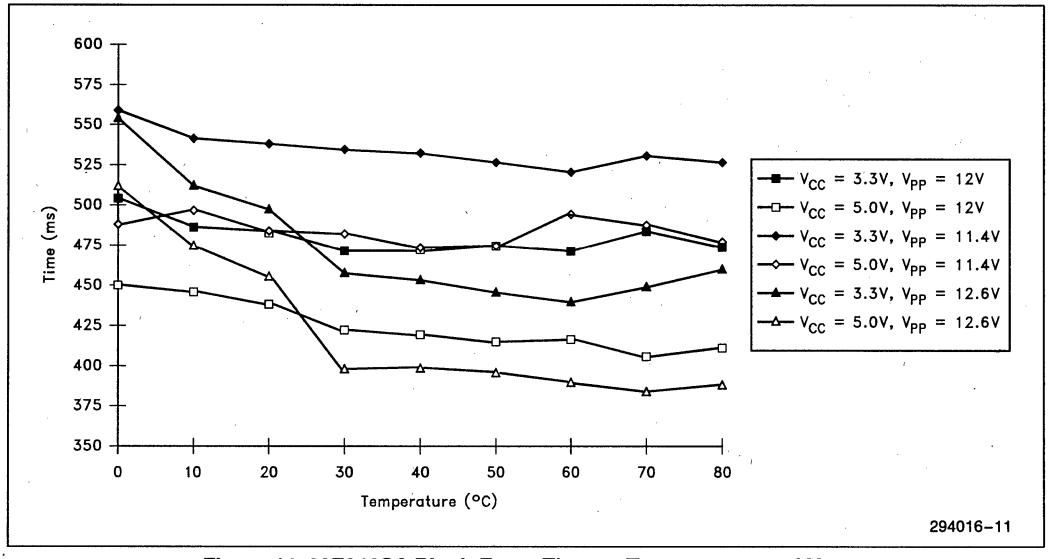


Figure 11. 28F016SA Block-Erase Time vs Temperature and V_{PP}. Block-Erase durations shorten at maximum V_{PP} and higher temperature.

Also, wear-leveling concepts, like those contained in Microsoft's FlashFile System and the embedded control code for Intel's Flash Drive product, can be used to periodically cycle stable data segments to other blocks, which ensures distributed usage and minimal variation in Block-Erase and Word-Write times across blocks. Consult Intel for more information regarding these concepts.

To bring a 1,000,000-cycle specification into perspective, it would take >10,740 hours to cycle every block on a single 28F016SA device 1,000,000 times, i.e., given blocks were continually erased and rewritten as fast as these operations could be performed. (This assumes byte-wide mode with 225 ns of overhead to initiate each programming operation. No Status Register Read times are included, and only one pulse is assumed for all 2,097,152,000,000 programming sequences. Erase is calculated at the 800 ms typical duration for all 32,000,000 sequences.) Of course this is not realistic usage-typical modus operandi for file-storage media is 80% reads vs 20% writes. Also, the average human user does not work 24 hours a day saving data files, just to rewrite them.

Cycling-induced hard failures do not occur until an intrinsic wearout mechanism, caused by electron trapping, becomes so severe that the allotted maximum number of internal WSM program or erase pulses cannot overcome their opposing potential. Low electric fields, advanced low-defect oxides and minimal oxide area per cell combine to greatly reduce oxide stress and prolong the probability of this failure.

From a performance perspective, electron trapping eventually pushes out write and erase times. Within the boundary conditions of the device, these times still conform to specification. Explicitly, hot electrons from programming trap in the tunnel oxide near the drain junction. This creates a negatively-charged barrier slowing CHE injection. Similarly, electrons trap near the source junction because of erase. Their negative potential makes F-N tunneling less efficient. Figures 12 and 13 show the result of these degradations over long-term cycling.

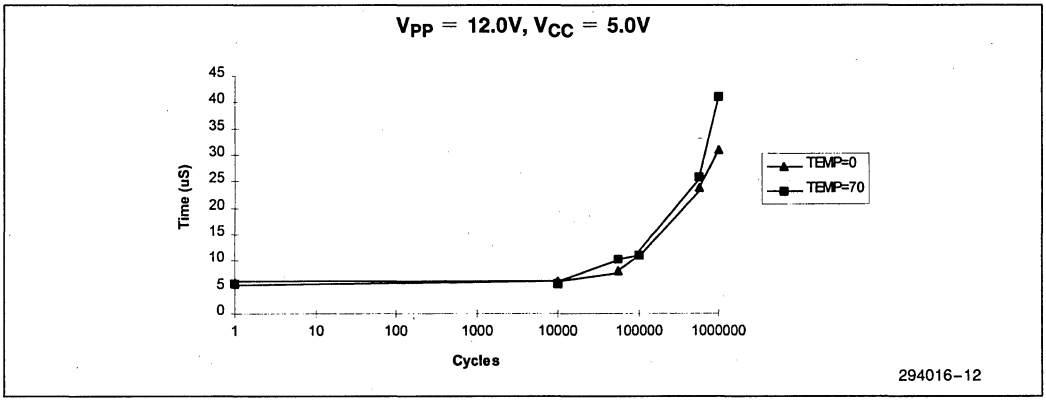


Figure 12. 28F016SA Average Word-Write Time vs Cycling. The effects of cycling on Word-Write times are negligible through 10,000 cycles. These times increase after 10,000 cycles, but Word-Write operations remain reliable up to 1,000,000 cycles.

9

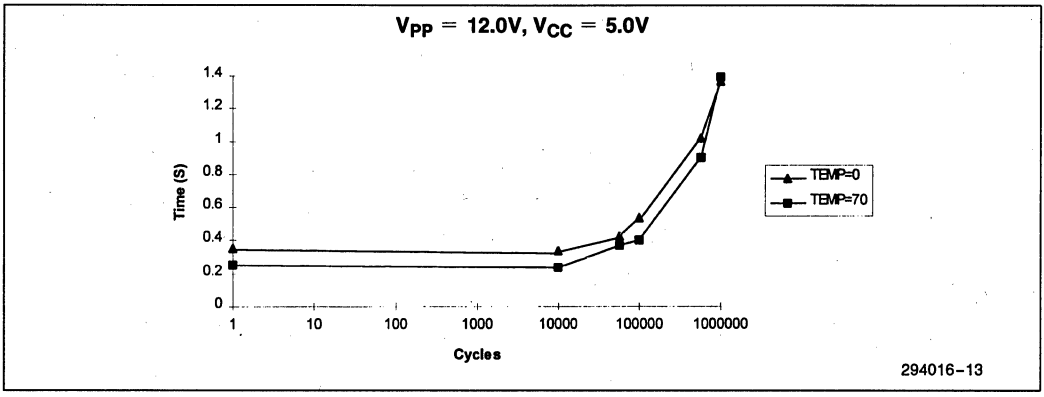


Figure 13. 28F016SA Average Block-Erase Time vs Cycling. Average Block-Erase times decrease over the first 10,000 cycles. These times increase as the cycle count nears 1,000,000, but Block-Erase operations continue to perform reliably to product specifications.

Program and erase V_t distributions are a critical factor in determining device reliability. The purity of Intel's ETOX IV thin-oxide formation technology, coupled with optimally-short WSM pulsewidth durations and a smaller oxide area under stress, reduces electron trapping. Reduced trapping behavior prolongs buildup of electrons near the drain and source junctions, keeping threshold distributions tight maintaining the integrity of the tunnel-oxide isolation layer.

The robustness of ETOX IV minimizes the negative effects of write/erase cycling. Write and erase times remain consistent over the first 10,000 cycles, and typically double as the device nears 100,000 cycles. For programming, this means the likelihood of a second 2.5 μs WSM pulse; and for erase, the requirement for double the number of 10 ms pulses.

Process Variation

Intel's process control of ETOX IV critical cell dimensions keep Data-Write and Block-Erase electrical characteristics on target with little variance. This capability assures manufacture of devices with consistent write/erase performance, making product operation predictable.

Statistical variation, inherent to any process, will cause internal cell dimensions to exhibit some small degree of deviation. The primary process variable affecting erase is tunnel-oxide thickness. Channel length (L_{eff}) has the largest impact on programming. Outgoing product testing ensures performance to specification regardless of these minor variances.

Electrical Testing

Electrical testing gives added value to Intel Flash Memory products. This elaborate testing provides valuable insight to device characteristics and ensures product longevity. Electrical test is performed at both wafer and package levels (Figure 14). All die pass through wafer sort. This testing involves a screen of the array for defects and verifies overall circuit functionality. In so doing, each flash cell on every die is checked for integrity and proper operation.

All product shipped proceeds through finished package test prior to mark. This level of tests verifies correct assembly, checks for defects made visible from burn-in stressing and guarantees operation to all timing specifications. The array and peripheral circuits are again checked for correct operation.

Supply and input voltage levels and ambient temperature (T_A) are cycled to worst-case minimum and maximum corners during testing to ensure outgoing product performs to those parameters. Healthy guardbands compensate for tester tolerances and tester-to-tester variations.

In addition to guaranteeing that each part meets specifications, electrical testing serves as a means for continuous data collection. This accrual of information allows for a meaningful and optimized evolution of process and design, contributing toward a very robust, manufacturable and cost-effective technology.

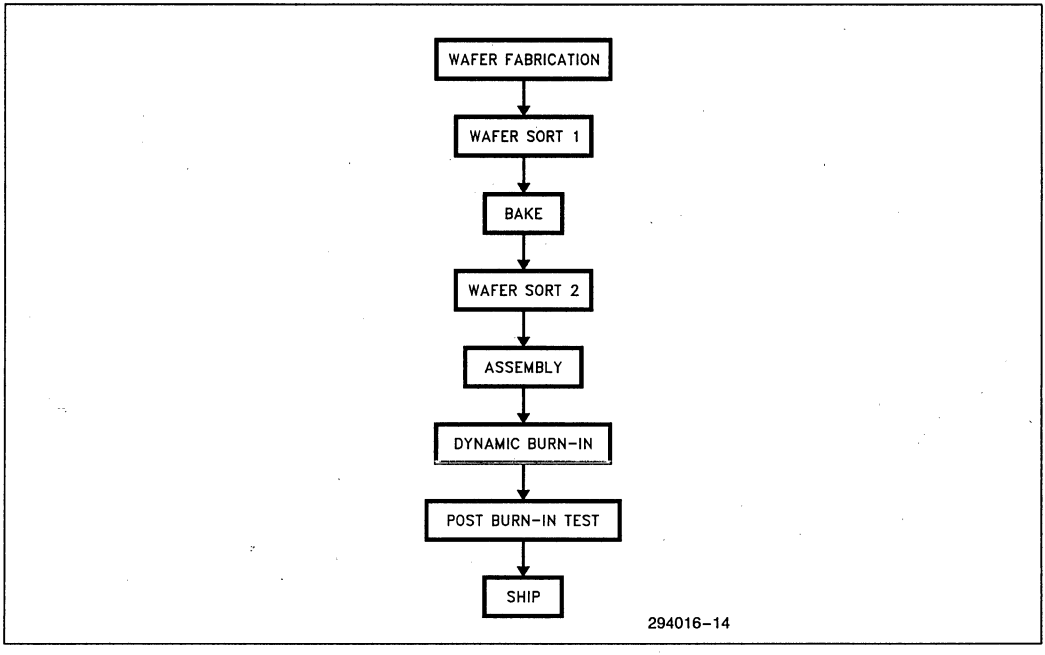


Figure 14. ETOX IV Electrical Testing Flow.
Electrical test and its associated data collection ensures product integrity and quality meet high levels, while providing valuable information to further enhance process control.

9

ETOX IV PROCESS-RELATED IMPROVEMENTS

This section begins with a Historical and Evolutionary preface, and expands on the following areas:

- Cell/Array Compaction
- Enhanced Contact Technology
- Improved Cell/Array Uniformity
- Write/Erase Performance
- Cycling Performance
- Read Access

History and Evolution

To put the scope of such an advanced flash memory manufacturing process into perspective, it would be helpful to understand the evolutionary and revolutionary steps involved to reach that point. Intel, the world's largest IC supplier, is also the most widely experienced manufacturer of semiconductor technologies. From the memory side, the company leverages a long history in a broad variety of different types and associated processes including: Static, Dynamic, Pseudostatic and Integrated RAMs in Bipolar, MOS (PMOS and NMOS) and CMOS configurations, MOS/CMOS Erasable and Bipolar PROMs, MOS ROMs, Magnetic Bubble Memories, EEPROMs and NVRAMs. Intel invented and commercialized DRAM and EPROM technologies via:

- the 1103, 1024 x 1-bit Dynamic MOS RAM in 1970
- the 1702, 256 x 8-bit UV-Erasable MOS PROM in 1971

The 1702, using a charge storage element referred to as a Floating-gate Avalanche-injection MOS (FAMOS) device, gave birth to Intel's EPROM product family. This device had a cell size measuring 909 μm^2 via 10-μm lithography. The upgrade 1702A-2 offered this product's fastest maximum access time of 650 ns via $+5V \pm 5\%$ on three V_{CC} inputs and one V_{BB} input, and $-9V \pm 5\%$ on V_{DD} and V_{GG}. The requirements for programming input and supply voltages, with respect to the three V_{CC} inputs held at 0V, were:

- V_{BB} held at $+12V \pm 10\%$
- V_{DD} and PGM pulsed to $-47V \pm 1V$
- V_{GG} pulsed to $-37.5V \pm 2.5V$
- Address input voltages at 0V for logic "1" and $-44V \pm 4V$ for logic "0"
- Data input voltages were 0V (cell remains unchanged) or $-47V \pm 1V$ to program a logic "1" (output high in read mode)

Programming time for this device was very slow relative to today's technology; it required approximately 64 seconds for the tiny 2048-bit array. A large part of this was due to the greater number of input pins requiring transition with longer setup, hold and pulsewidth durations. However, the biggest factor affecting throughput was that the programmer had to scan through all addresses in ascending binary sequence 32 times (i.e., a data location could not be pulsed 32 times before moving up to next byte).

Since those early days, substantial advances have been made in the manufacturing process and architectural design resulting from a strong commitment to continuous test data collection and process monitoring, capital expenditure to maintain equipment and facilities on the forefront of leading-edge capabilities, and extensive R&D spending to lay the groundwork for today's technologies. These advancements made possible Intel's final EPROM development vehicle, the 8,388,608-bit 0.8 μm 27C800 in late 1990. This device, with a cell size of 6.5 μm^2, advanced then current EPROM technology to the state-of-the-art for read access and programming. Advanced array layout design and improved read path technology enabled an 85 ns maximum t_{ACC} from a single $+5V$ V_{CC} requirement. A theoretical array programming time of 6 seconds resulted from a simplified algorithm flow (very simple relative to the 1702 or 2708) with individual byte pulses at 10 μs, and address/data setup and hold times in the 400 ns and 1 μs range. Voltage requirements (referenced to the GND pin) were:

- V_{PP} at $12V \pm 0.25V$
- V_{CC} at $5.0V \pm 10\%$ during programming and $6.25V \pm 0.25V$ for margined verification
- Address, data and control inputs at CMOS/TTL-compatible voltages

In retrospect, early EPROMs like the 1702 and 1702A were constructed with p-channel MOS transistors, and had minimum-design-rule geometries of 10 μm. The first major technological milestone came about in 1975 when EPROMs transitioned to n-channel devices. Intel's 1024 x 8-bit 2708 dramatically improved device read performance because of the inherent higher speed of n-channel charge carriers (electrons) in silicon.

The 2708, which was manufactured on 6 μm technology, required three power supplies for read:

- V_{CC} at +5V ±5%
- V_{BB} at −5V ±5%
- V_{DD} at +12V ±5%

Its 2708-1 version was specified at 350 ns maximum t_{ACC}. Programming for this device, which was specified at 100-seconds typical for all 8 Kbs, allowed use of the same $V_{IL/IH}$ TTL-level address/data input signals required during reads. The following additional input voltages (referenced to the GND pin) were required:

- CS#/WE input held at +12V ±5%
- PGM input pulsed to +26V ±1V

The 2716, introduced in 1977, evolved the technology to a single +5V ±5% V_{CC} supply requirement for read. Access time though, remained at 350 ns maximum for the fastest product offering. The 2716 also incorporated a dedicated programming supply input, V_{PP}, requiring +25V ±1V, with address and data set-up and hold time minimums reduced to 2 μs.

It wasn't until 1980 though, when Intel's 2764, the first EPROM manufactured on the company's patented HMOS-E (High-Performance n-channel MOS) technology, that read performance made a large gain by achieving 200 ns. This device incorporated a 159-μm^2 cell via 4-μm geometries. The next HMOS-E device, a 2732A (stepping of the 2732) reduced cell size to 100.6 μm^2 via 3-μm geometries.

Since then, HMOS-E progressed to HMOS II-E and HMOS III-E, with dimensions shrinking from 4/3 μm to 2.0/1.5 μm to 1.2 μm. These process evolutions enabled shorter programming pulsewidth durations and address/data setup and hold times, as well as a reduction in the V_{PP} programming voltage. New algorithm flows could be developed, enhancing production programming throughput. Also, coupled with architectural design advances in array layout, decoding and sense amplifier circuitry, read performance continued to improve to 150 ns.

CMOS circuits, implemented for the periphery of the device, were first used on the 27C64. This was the next major milestone for EPROM technology, as it provided significant reduction to device power consumption. This advancement ushered in Intel's next patented EPROM process CHMOS II-E (Complementary HMOS), which fostered geometries of 1.5 μm and 1.2 μm. CHMOS III-E produced a geometric reduction to 1.0 μm, and 0.8 μm was realized on what would have been CHMOS IV-E if Intel had continued to stay in the EPROM market. It was also becoming very apparent, at this point, that migration toward smaller geometries would make floating-gate discharge (erase) via ultra-violet light exposure increasingly longer and more difficult to achieve.

Intel's first Flash Memory device, the 57F64, was a 8K x 8 Flash EPROM manufactured on the company's CHMOS II-E process. This device, when introduced in early 1988, leveraged a three-year manufacturing base built on the CHMOS process. It laid the groundwork for the first ETOX I product, the 27F64 flash memory, which was soon followed by the higher-density 27F256, 28F256P1 and 28F256P2 devices.

Intel's 28F256A, 28F512, 28F010, 28F001BX and 28F020 flash memories, which were introduced in 1990 and 1991, are manufactured on the ETOX II process, which itself had knowledge gained from 1.0-μm CHMOS III-E technology. ETOX III devices such as the 28F002BX, 28F200BX, 28F004BX, 28F400BX and 28F008SA, which were introduced in 1992, leveraged the earlier 0.8-μm development work performed on the 27C010A and 27C800. The 28F010, 28F001BX and 28F020 have since transferred to 0.8-μm manufacture.

This long history of experience has allowed Intel to travel further along the semiconductor memory learning curve than any other flash manufacturer in the world. Continuous improvements in process control have culminated in the highest levels of quality, reliability and manufacturability which are inherent to the company's ETOX IV process. The most notable sign of this quality and reliability is in the cleanliness and integrity of the ETOX IV tunnel oxide. All areas of performance for a flash device, such as consistent timings, data retention and cycling durability depend, to a great extent, on the tunnel oxide.

ETOX IV maintains an advanced scaled substrate EPI thickness to practically eliminate product latch-up, and uses double-metal technology to carry bitline data and strap the wordline. These process advancements aid cell/array compaction and enhance device read/write times. Array compaction, in turn also boosts read/write performance. Double-metal technology, first used on ETOX III, requires improved planarization processing. A desirable by-product of improved planarization is enhanced moisture performance.

ETOX III achieved a very high level of planarization through several of its process improvements. The enhanced contact materials and methodologies of ETOX IV aid in making the array structure even more planar. Additionally, tighter cell dimensions proliferated across the entire array enhance planarity by making it more consistent.

Greater structural uniformity and improved step coverage enhance process manufacturability, improve product reliability, and make program and erase threshold distributions and read/write timings more consistent. Also, ETOX IV's smaller oxide area per cell, coupled with extremely low defect density, provide lasting integrity for cycling and data retention.

Cell/Array Compaction

Not too many years ago, it was doubtful whether semiconductor minimum dimensions would ever scale below 1.0 μm. Through roughly six years of Intel Flash Memory development, ETOX process geometries have shrunk from 1.2 μm on a device like the 28F256-P1C2 to 0.6 μm on the 28F016SA. This rapid advance in reducing minimum-design-rule geometries not only exhibits superior technical prowess, but also increases cell density per unit area, thus keeping flash memory cost-per-bit learning on track with prior expectations.

The shrinking of minimum dimensions also enables the capacity to cost-effectively incorporate higher levels of logic integration, while maintaining or reducing the periphery-to-array transistor ratio. This logic integration brings the control functions for flash memory operations on the chip. Also, new features can be cost-effectively added into designs (28F016SA as example).

In addition to higher integration, array compaction allows a means for packing dense memory devices into ultra-small packages like 56-ld TSOP (14-mm-long x 20-mm-wide x 1.2-mm-thick Thin Small-Outline Package). This small package gives the versatility to build compact high-density modules and systems like a 40-MB Flash Memory Card with form and fit to PCMCIA 68-pin type-I mechanical specifications.

Enhanced Contact Technology

Ever-shrinking geometries mandate continuous improvement to existing production methodologies and materials, and development/use of new capabilities. These improvements and developments allow compaction not possible via minimum-design-rule reductions only.

The enhanced contact technology of ETOX IV is such an example of new development and continuous improvement, enabling the tighter placement of array cells via metal-1-to-silicon contact hole minimization. Step coverage and contact resistance improve via these advances, enhancing yields, reliability and performance.

Improved Cell/Array Uniformity

Large flash memory arrays require uniform cell structures to maintain tight program and erase V_t distributions. Consistent V_t distributions are critical for reliable device operations, especially Block Erase.

ETOX IV employs new manufacturing techniques that improve cell, and therefore array layout uniformity. In addition to improved threshold distribution, this uniformity provides:

- more consistent read, write and erase timings,
- increased product reliability,
- a higher degree of cell/array compaction,
- a more planar silicon formation,
- and greater manufacturing yields.

Write/Erase Performance

Data-Write and Block-Erase timings are reduced from previous product generations via the combined effects of geometric scaling, enhanced cell/array dimensional uniformity and optimization of WSM pulses, support circuitry and algorithmic flows. Additionally, new device features like multiple command queuing, a Block-Erase tagging mechanism and fast data-in caching buffers improve write and erase performance at the system level.

An externally-controlled timeout of 100 μs programs a byte on the 1.5-μm 28F256-170P1C2. An internally-controlled pulsewidth of 2.5 μs programs a word or byte on the 28F016SA. 28F016SA erase, which includes WSM pre-erase conditioning of all block bits (64-KB/32 KW-block), typically requires 600 ms. A combination of 4-second typical chip-program and 1-second typical chip-erase times is necessary to net the same erase result on the 32 KB 28F256-170P1C2, whose entire array is half that of one 28F016SA erase block.

Cycling Performance

Cycling durability has made tremendous gains from early first-generation devices, such as the 28F256-170P1C2 with a 100-cycle specification. Current products, such as Intel's ETOX III 28F008SA, are registering above 1,000,000 cycles. Experiments with these units have been conducted to prove their reliability. These parts typically do not produce hard failures, instead they see program and erase timings push out.

ETOX cycling longevity has progressed due to continued improvements in tunnel-oxide quality. As oxide layers become cleaner and more consistent, cycling-induced electron trapping minimizes. Reduced electron trapping helps keep V_t distributions tight. Additionally, reduced electron trapping improves program and erase pulsing efficiencies. This improved efficiency lessens the number of pulses required for a given operation to succeed, which in turn also reduces electron trapping. Since trapping is the primary source of program/erase time pushout (i.e., more pulses required), higher quality oxides reduce this degradation and extend cycling performance.

Improved source-to-floating gate coupling consistency via enhanced cell/array uniformity also extends cycling longevity by lowering the number of erase pulses required. Additionally, the Intel ETOX cell, in combination with optimal WSM control, typically requires a very low number of pulse repetitions to achieve a program or erase state (typically one pulse for programming). This also lessens the pulse count, extending cycling livelihood.

Improvements in isolation, both from cell to cell and around the floating gate, have practically eliminated charge retention and data disturb concerns. Contrary to competitor reports, parts with more than 1,000,000 cycles retain data as well as devices with much less cycling.

Read Access

Intel Flash Memory architecture has evolved by leaps and bounds since its inception. Read access time (t_{ACC}) has made significant gains, while maintaining very high levels of noise immunity. The 27F64's fastest speed bin was 150 ns with $\pm 5\%$ V_{CC}, while the 28F016SA will offer a 5.0V 70-ns/80-ns version with $\pm 5\%/10\%$ V_{CC} tolerance and a 3.3V ± 0.3V 120-ns version.

Double metal, introduced on ETOX III, has been the biggest contributing factor toward improving read access times. This enhancement reduces wordline and bitline resistance, thereby enabling faster cell turn-on and sensing paths during read operations. Continually-improved circuit designs also contribute to shorter read timings. Again, cell and array compaction enables inclusion of new/additional circuitry that enhance read performance, and will also allow future 3.3V parts with access times much faster than their 5.0V predecessors.

FLASH vs. OTHER SEMICONDUCTOR MEMORY TECHNOLOGIES

Intel's scaling advances in flash memory manufacturing and design provide optimal cell/array compaction. In roughly twenty-two years, Intel non-volatile memory density has grown from 2,048 bits to 16,777,216 bits, a factor of 8192x. Figure 15 compares other memory types to show relative density progression. The fast ramp in ETOX flash memory density results from its similarity to EPROM.

Figure 16 illustrates the relationship between cell sizes of different memory types and minimum geometries. As dimensions scale, certain memory types become cell-size limited (i.e., some components cannot shrink proportionally). The memory cost-per-bit learning curve shows flash in a strong position. This curve, shown in Figure 17, reflects how Intel's experience reduces cost for increased memory density.

Since the late 1980's, a new memory sub-system has arrived on the market offering an alternative to high-density file system media. Intel's Series 2+ Flash Memory Cards take advantage of the 28F016SA and its third-generation architecture to provide card densities of up to 40 Mbytes and new functionality. This relatively new technology offers a solid-state file system (Figure 18) that will double in density with new ETOX generations.

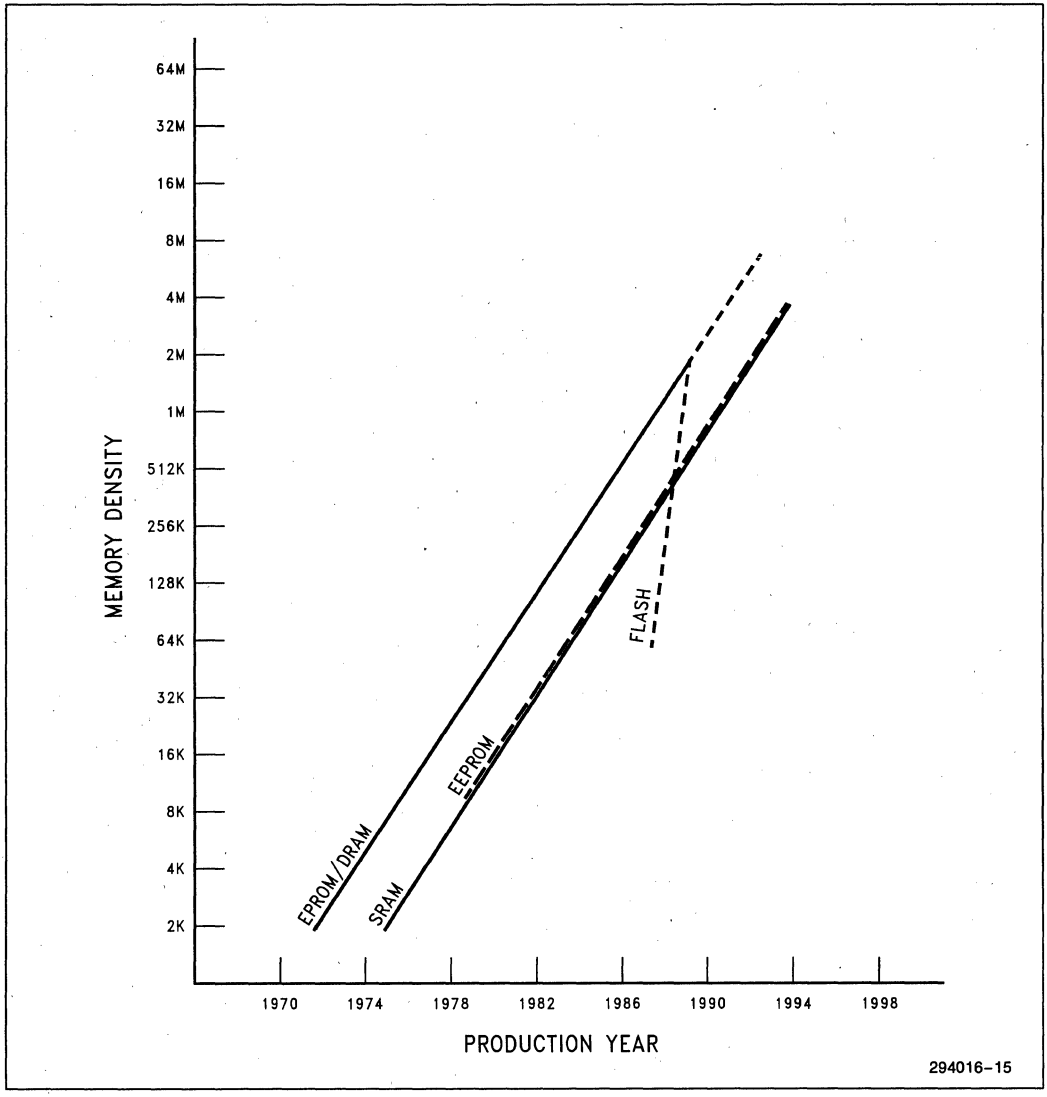


Figure 15. Memory Evolution

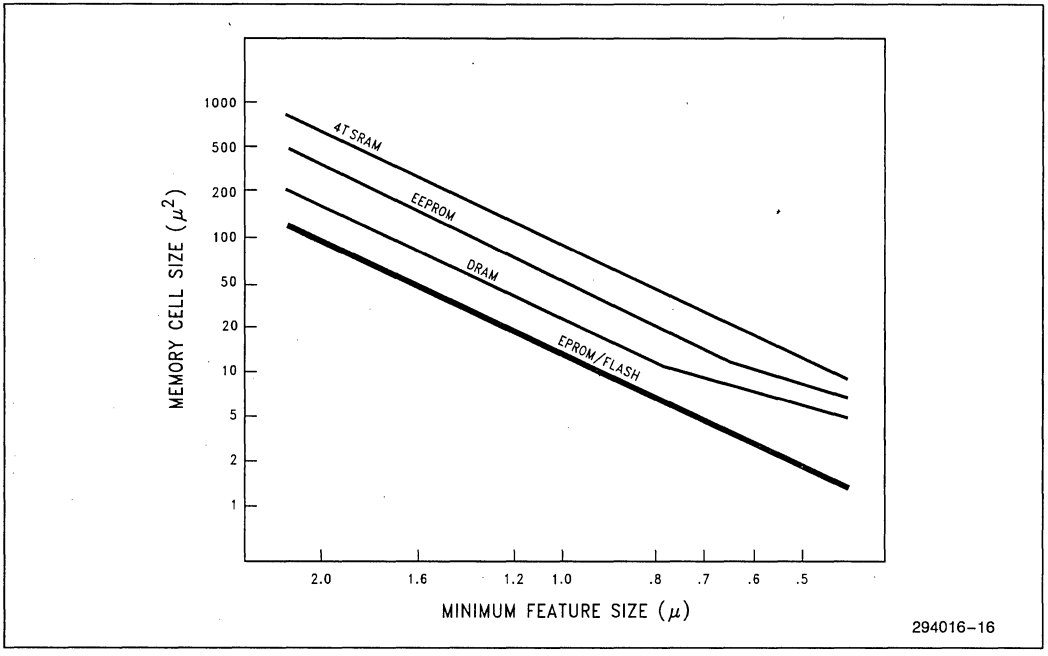


Figure 16. Memory Cell Size Trends

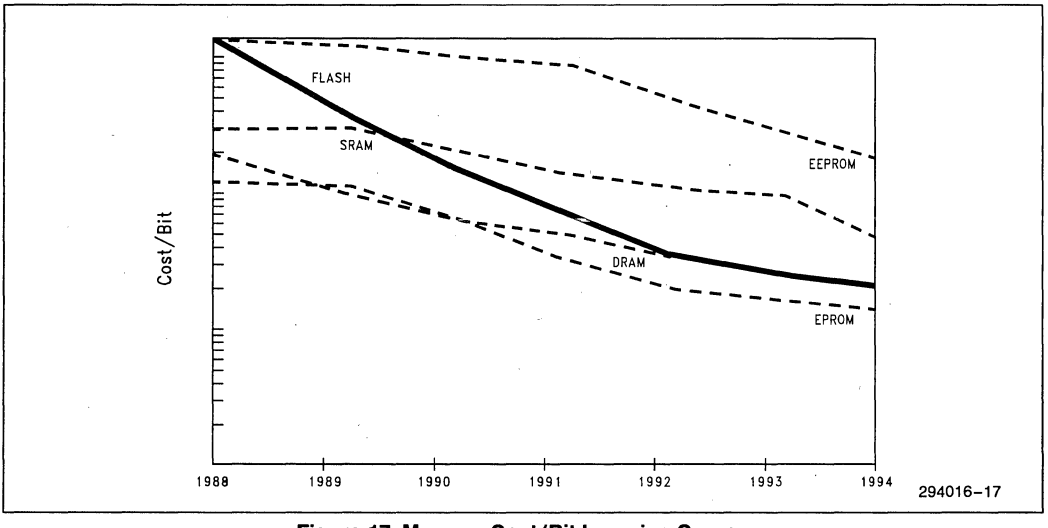


Figure 17. Memory Cost/Bit Learning Curves

9

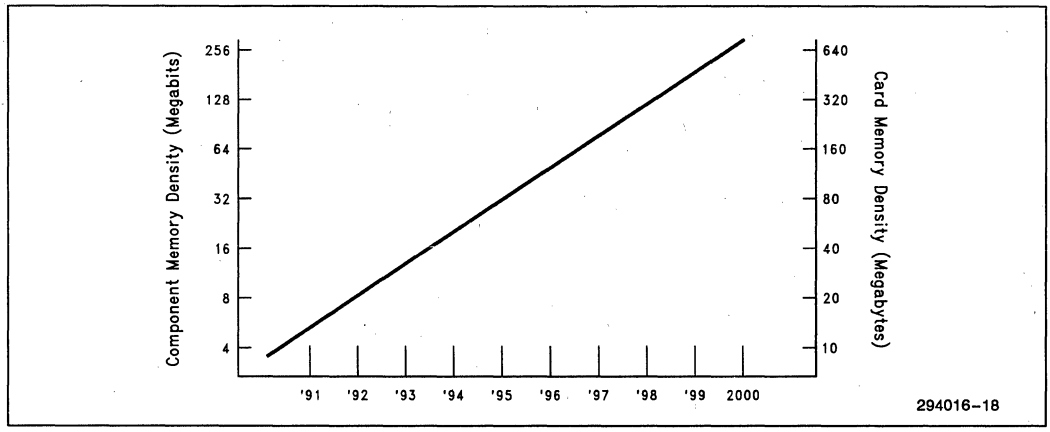


Figure 18. ETOX Component and Memory Card Density Over Time

SRAM and DRAM

SRAM and DRAM have fast read/write speeds. Both though, are volatile memories requiring continuous power to retain data. Standard SRAMs (Figure 19) require four to six transistors for each flip-flop cell. This greatly reduces memory capacity per unit area, raising product cost for a given density.

DRAM requires constant refresh of its capacitor-like storage mechanism (Figures 20 and 21) due to leakage currents and read operations. Charge storage requirements limit size reduction of this capacitor, which in turn limits memory array compaction. With smaller geometries, DRAM cell structures become increasingly complex requiring additional and more difficult manufacturing steps.

Most DRAMs require read parity. This additional bit is needed for two reasons: First, alpha-particle strikes can disturb cells via ionizing radiation, with resulting loss of data. Second, during reads, only a small voltage differential is available for sensing because the cell's storage mechanism capacitively shares its charge with the bitline through the select transistor. This low voltage differential can also be influenced by nearby bitline voltages and device noise.

Parity bits are not required on floating-gate technology because the substrate is electrically isolated from the charge storage mechanism. Unlike DRAM, floating-gate charge determines cell V_ts, which in turn controls bitline voltages. This allows flash memory read sensing to easily detect cell V_ts.

Most DRAMs require high active power consumption. Charge storage requirements and read signal strength constrain DRAM cell compaction. Low-power ETOX IV flash memory has a simple single-transistor cell with only minor scaling limitations through the year 2000. This results in a mainstream memory that does not need power to retain data.

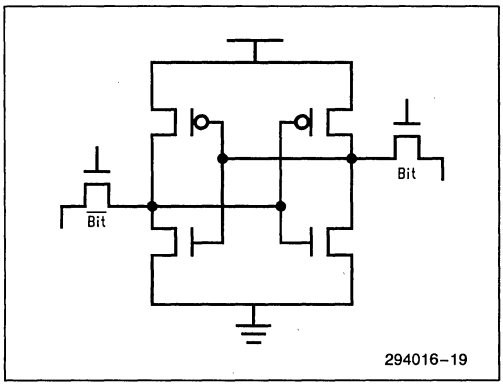


Figure 19. Six-Transistor SRAM Cell Schematic

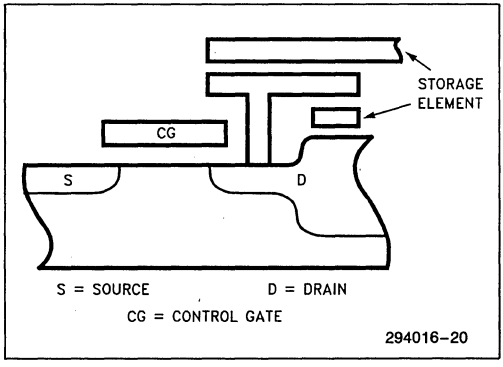


Figure 20. Stacked DRAM Cell (Side View)

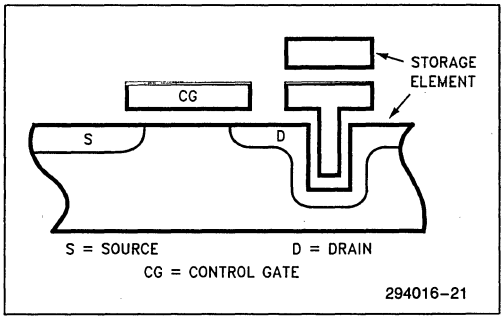


Figure 21. Trenched DRAM Cell (Side View)

EPROM

Intel's prior EPROM technology and Flash Memory cells share a common stacked-gate profile (Figure 22), with two basic differences relating to their respective erase mechanisms. EPROM requires ultraviolet light to erase, flash erases electrically. Electrical erase is possible because flash has a thinner cell oxide to allow F-N tunneling, and a graded source diffusion to improve source-to-floating-gate coupling and prevent breakdown during erase.

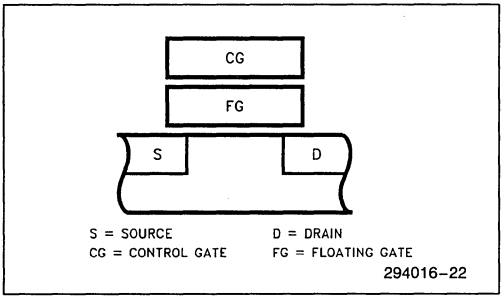


Figure 22. EPROM Cell (Side View)

EPROM technologies migrating to smaller geometries will find floating-gate discharge (erase) via UV exposure increasingly difficult. One problem is that metal bitline widths cannot reduce proportionally with process technology advancements. EPROM metal-width requirements limit bitline compaction in order to maintain low resistance. As such, they reduce the amount of high-energy photons that otherwise would reach charged cells. Additionally, as dimensions scale, airborne (and other) particles that were once insignificant, now become a factor as they also obscure photons. Therefore, EPROMs built on sub-micron technologies will face longer UV exposure times.

Intel's ETOX IV technology employs a second-metal processing step to strap wordlines for improved read performance. This advance is not likely to appear on EPROM because it would block even more UV light. Since flash memory electrically erases, it eliminates these concerns. Moreover, flash electrical erasure eliminates the UV window and its associated cost. This allows ready acceptance of the most advanced and innovative plastic surface-mount packaging solutions.

EEPROM

Conventional two-transistor EEPROM cells limit layout density. In addition to the main cell shown in Figures 23 and 24, a second transistor works as a bit select. This extra transistor and decoding circuitry are required for single byte program and erase capability. Technology design requirements make EEPROM cell, like triple-poly, significantly larger than flash. Typical EEPROM technologies are more complex, making wafer manufacturing difficult and expensive.

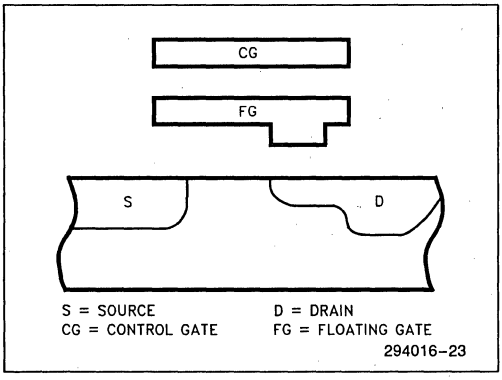


Figure 23. Flotox EEPROM Cell (Side View)

Because of their traditional application, EEPROMs use a very high internal voltage (17V to 30V) to achieve fast program and erase times. These high voltages and resulting electric fields cause cell oxides to breakdown, shortening cycling life and degrading cell thresholds. Additionally, this high voltage stresses periphery transistors. Intel's Flash Memories are more akin to EPROM; both use a significantly lower voltage around 12V.

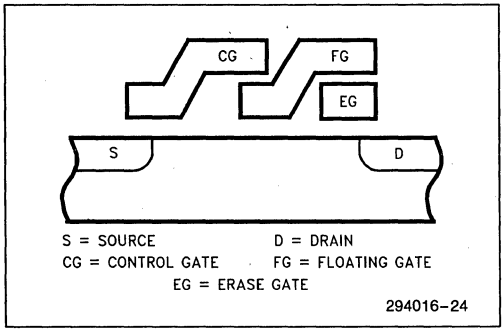


Figure 24. Triple-Poly EEPROM Cell (Side View)

PROCESS/DESIGN TRADEOFFS TO ACHIEVE THE PERFECT FLASH MEMORY

There are two main differences in flash memories today. The first is in the structure of how the individual flash cell is built. Examples are Intel's stacked-gate approach and SunDisk's triple-poly flash cell. The second difference is in the architecture of the array. Array layout not only affects the structure and complexity of the chip, but it also influences the methods used to write, erase and read the device, and the subsequent performance of these operations. This section describes several of today's prominent flash memory technologies, and discusses various characteristics for each.

NOR Flash

There are several types of NOR flash in development today, most of which use structures similar to Intel's stacked-gate ETOX cell. The defining characteristic of NOR technology is array layout (Figure 25). In this implementation, many flash cells are connected, via their drains, to a common column, or bitline. Many bitlines are grouped to make up individual I/Os. A select line, or wordline, serves as the control gates for a row of flash cells, one on each bitline. This layout allows for many cells, usually a byte or word, to be accessed in parallel, thereby providing for fast random read/write access performance. Different NOR technologies vary mainly in erase methodology.

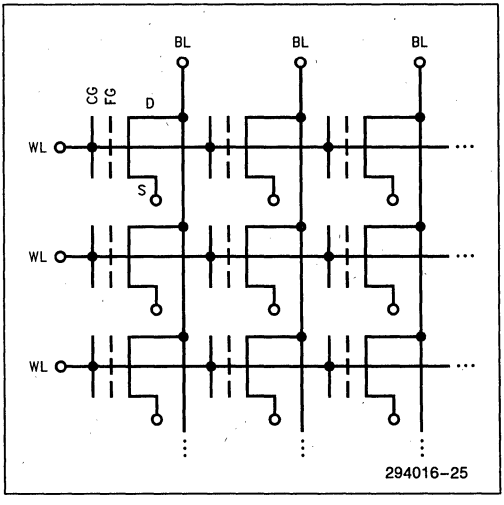


Figure 25. NOR Flash Array

ETOX Flash

Intel's ETOX NOR technology uses a High-voltage Source with grounded-gate Erase (HSE) methodology, as described earlier in this report (Figure 7). A paper published by Toshiba Corp. in 1992 entitled, *"Comparison of Current Flash EEPROM Erasing Methods: Stability and How to Control"*, describes Intel's HSE as "the most stable scheme for the control of erasing speed and erased threshold voltage distribution." It also concluded that certain tools are needed for "suppressing the erased-V_t distribution width" in other methods of erase, such as Negative Gate-bias Erase (NGE).

Negative Gate-Bias Erase (NGE)

The method of erasing flash cells via NGE is inherently more complex than Intel's HSE approach. There are two main variations of this scheme. The first is a Negative Gate with positive Source Erase (NGSE) method. NGSE requires a large negative voltage on the cell's control gate while placing an intermediate positive voltage on the source. Advanced Micro Devices, Inc. uses this approach with voltages as seen in Figure 26. The second is a Negative Gate with positive Channel Erase (NGCE). This practice is being considered by NEC Corp. by placing voltages as seen in Figure 27 on the cell.

Routing negative voltages throughout the device requires a higher degree of isolation in the periphery than needed for HSE devices. Consequently, the peripheral circuitry consumes a larger percentage of total die area. In addition to increased isolation, a triple-well process or large p-channel devices are necessary to switch the negative voltages onto cell gates. An on-chip charge pump must generate this negative voltage. These three requirements increase die size, thus inflating the product's cost.

In general, a more complex manufacturing process is required to produce flash memory devices using NGE. Small statistical variations in manufacturing parameters can cause threshold voltage instability in these devices. Without the use of advanced tools and close process monitoring, this complex process can lead to inherently less reliable and inconsistent flash memory devices.

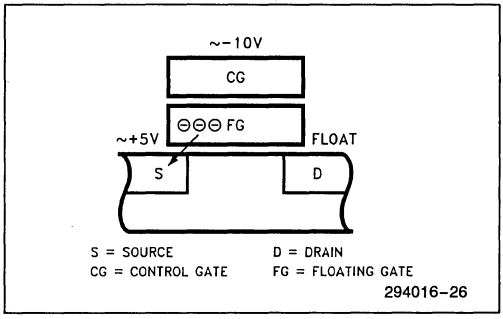


Figure 26. NGSE Cell Erase Voltages

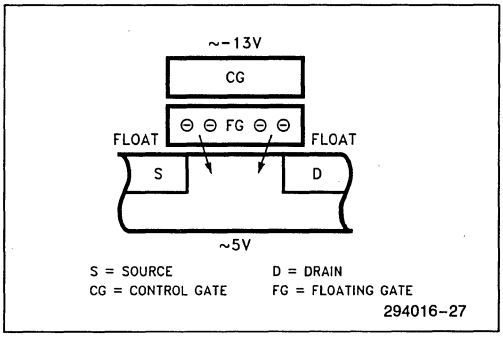


Figure 27. NGCE Cell Erase Voltages

NAND Flash

NAND flash memories, such as those produced at Toshiba Corp., use a cell similar to NOR, but with a distinctively different array layout (Figure 28). NOR cells are accessed in parallel, while NAND access is serial. This serial access unfortunately leads to very slow random read timings. Toshiba's 16 Mb NAND has random read timings that are more than two orders of magnitude longer (15 µs vs. 70 ns) than those available via a NOR-layout approach.

Also, if the data-read requirement is greater than 256 bytes in length, additional delays occur. These additional delays happen on the first memory access when page boundaries are traversed (i.e., the Toshiba 16 Mb NAND array is decoded into 256-byte pages; switching to read from another 256-byte boundary causes a 15 µs delay to valid data-out.

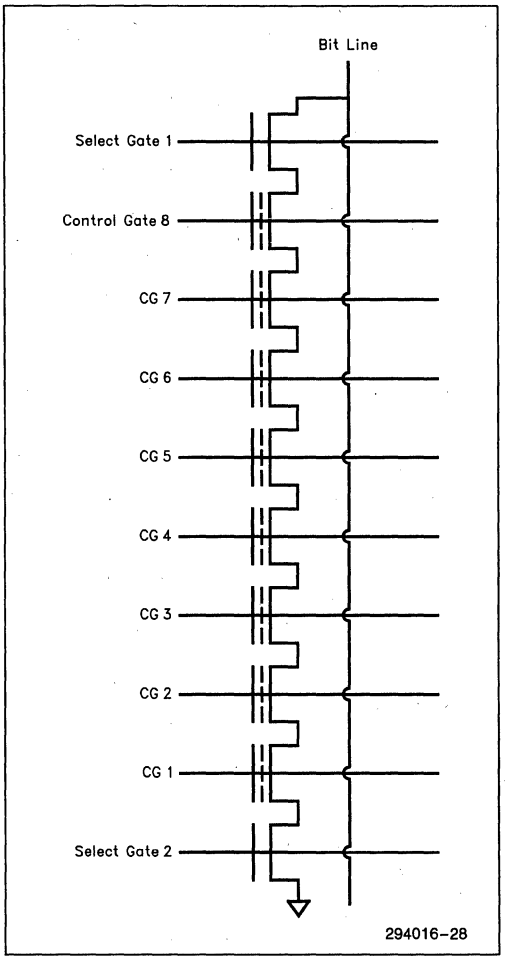


Figure 28. NAND Flash Array

NAND program and erase methodologies differ from those of NOR devices. Toshiba Corp. employs F-N electron tunneling for both operations. A very high voltage (20V) is applied to the control gate during programming, and to the substrate for erasure (Figure 29). This high voltage amplifies reliability risks in both cases. Tunnel-oxide breakdown is more probable than with Intel's ETOX technology. In fact, Toshiba Corp.'s flash memories compensate for these concerns by incorporating extra bytes in each page for error correction.

NAND's serial scheme allows tighter compaction in the array because its architecture only requires one contact for each string of 8 or 16 cells, vs. one contact per each cell pair in NOR devices. The downside for NAND is that this gain is lost due to an overall drop in array efficiency. This loss results from increased decoder size and complexity necessary to handle the serial array structure. Also, large charge pumps and extra isolation in the periphery are needed because of the requirement for very high write and erase voltages. Besides impacting reliability, high voltage isolation also induces fixed limitations to the future scalability of this technology.

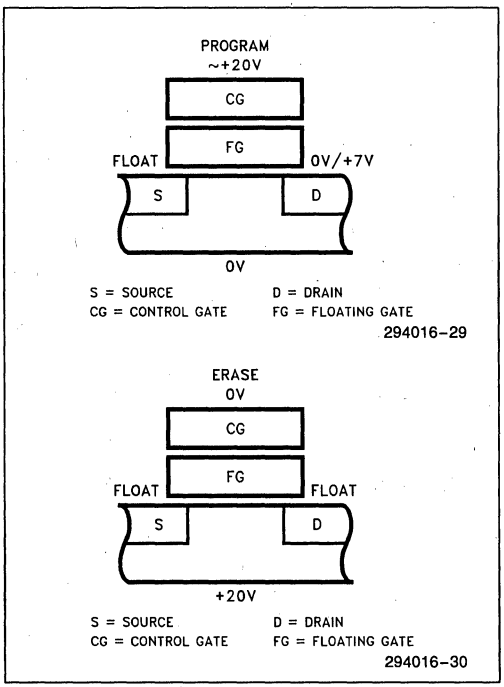


Figure 29. NAND Cell Program and Erase Voltages

Triple-Poly Flash

The name of this technology reveals its cell structure difference from the ETOX double-poly stacked gate. See Figure 30 for a schematic representation of this cell and array configuration. Although the theory behind triple-poly is sound, manufacture is very difficult. As such, SunDisk Corp. uses extensive error correction and redundancy in their devices to provide yield and reliability enhancement.

This technology incorporates a poly-1 floating gate, storing charge in much the same way as an ETOX cell. Programming, via CHE injection, proceeds by placing 12V on the poly-2 control gate and 7V on the drain (Figure 31). The source and erase gate are at 0V.

The third poly layer is used during erase operation. Poly-to-poly tunneling is invoked by bringing the poly-3 erase gate to voltages ramping between 12V and 22V. This high voltage prevents minimum design-rule scaling below certain limits, and increases the likelihood of oxide wearout due to a high field stress across the small oxide area.

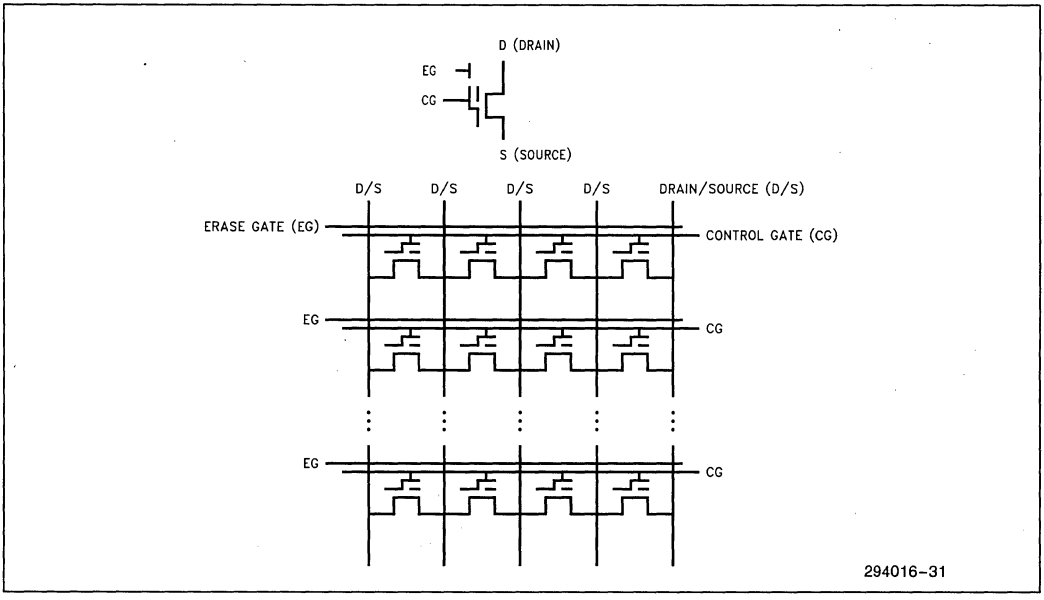


Figure 30. Triple-Poly Flash Cell Schematic and Array Configuration

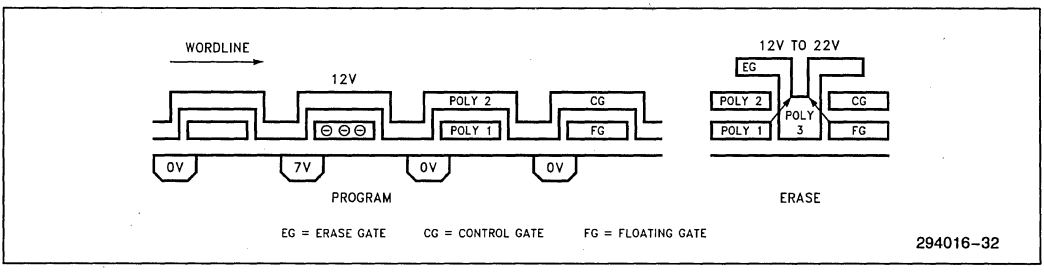


Figure 31. Triple-Poly Cell Program and Erase Voltages

9

SUMMARY

Intel's technological advances result in flash memory products that are more efficient, more reliable, less expensive and higher performance. Finer geometries, double-metal technology and EPI wafer processing allow considerable gains in memory array compaction. This compaction nets improved write/erase performance and lower product cost.

New ETOX IV manufacturing steps and contacting technologies also contribute to cell compaction and improved reliability. Extremely high cycling endurance results from the superior quality of Intel's ultra-pure thin tunnel oxide and the electrical characteristics of internal program and erase operations.

Cycling, voltage and temperature exhibit only a small influence on Data-Write and Block-Erase speeds. Products built on Intel's CMOS ETOX IV Flash Memory technology require very low power consumption during Data-Write, Block-Erase, Read, Sleep and Standby modes.

Intel Flash Memory products designed on ETOX IV will satisfy many different applications. The Thin Small-Outline Package (TSOP) provides high memory density in an extremely small footprint. Some applications include memory cards, solid-state drives, resident code (O/S and application) and file storage, data acquisition and embedded code storage.

The solid-state nature of flash provides far superior ruggedness over mechanical rotating media. With blocking, applications can perform erase as a background task to optimize system performance.

Today, Intel's technological advances in flash memory are driving cost to parity with DRAM. This steep decline in the price learning curve enables new classes of systems and system architectures.

OTHER REFERENCES

Related documents of interest to readers of this engineering report:

Data Sheet: 28F016SA 16-Mbit (1 Mbit x 16, 2 Mbit x 8) FlashFile™ Memory (Order No. 290489-002)

Data Sheet: 28F032SA 32-Mbit (2 Mbit x 16, 4 Mbit x 8) FlashFile™ Memory (Order No. 290490-002)

28F016SA 16-Mbit FlashFile™ Memory User's Manual (Order No. 297372)

AP-362: Implementing Mobile PC Designs Using High Density FlashFile™ Components (Order No. 292097)

AP-375: Upgrade Considerations from the 28F008SA to the 28F016SA (Order No. 292124)

AP-377: the 28F016SA Software Drivers (Order No. 292126)

AP-378: System Optimization using the Enhanced Features of the 28F016SA (Order No. 292127)

REVISION HISTORY

Number	Description
002	Updated Cycling data in Figures 12 and 13 to 1,000,000 cycles. Removed placeholders for Figures 14, 15, 16, and 17 and associated text. Contact Intel for more information.

intel®

10

Flash Memory SmartDie™ Products

28F010 1024K (128K x 8) FLASH MEMORY

SmartDie Product Specification

- **Flash Electrical Chip Erase**
  - **1 Second Typical Chip Erase**
- **Quick-Pulse Programming Algorithm**
  - **10 μs Typical Byte Program**
  - **2 Second Chip Program**
- **100K Erase/Program Cycles Typical**
- **12.0V ±5% V_{PP}**
- **High-Performance Read**
  - **90 ns Access Time**
- **CMOS Low Power Consumption**
  - **10 mA Typical Active Current**
  - **50 μA Typical Standby Current**
  - **0 Watts Data Retention Power**
- **Integrated Program/Erase Stop Timer**
- **Command Register Architecture for Microprocessor/Microcontroller Compatible Write Interface**
- **Noise Immunity Features**
  - **± 10% V_{CC} Tolerance**
  - **Maximum Latch-Up Immunity through EPI Processing**
- **ETOX™ III Nonvolatile Flash Technology**
  - **EPROM-Compatible Process Base**
  - **High-Volume Manufacturing Experience**
- **Intel SmartDie Product**
  - **Full AC/DC Testing at Die Level**
  - **0°C to + 80°C (Junction) Temperature Range**
  - **Available in 90 ns Access Time Only**
- **Reference the 28F010 1024K (128K x 8) CMOS Flash Memory Data Sheet, #290207, for Device Parameters and Architectural Features**

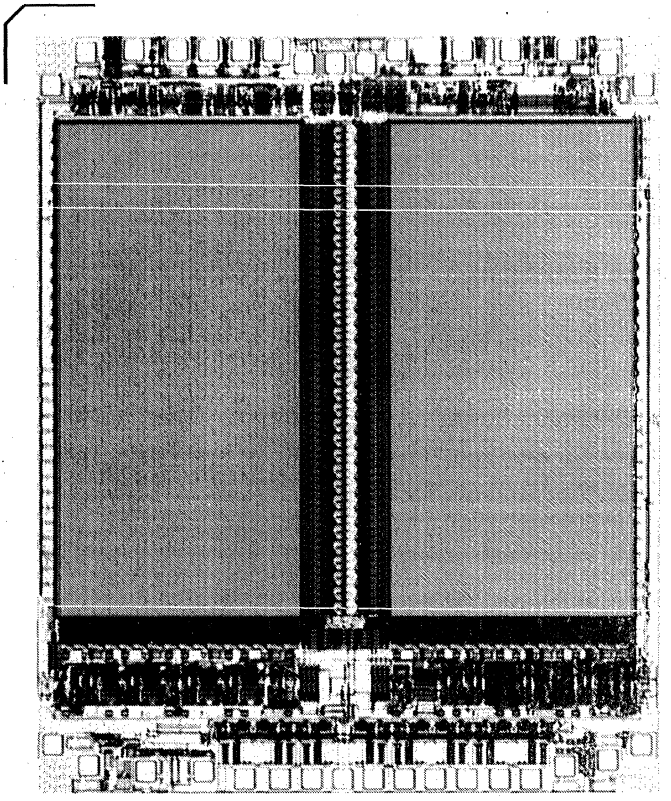
10

NOTICE: This data sheet contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

REFERENCE INFORMATION: The information in this document is provided as a supplement to the Standard Package Data Sheet on a specific product. Please reference the Standard Package Data Sheet/Book (Order No. 290207) for additional product information and specifications not found in this document.

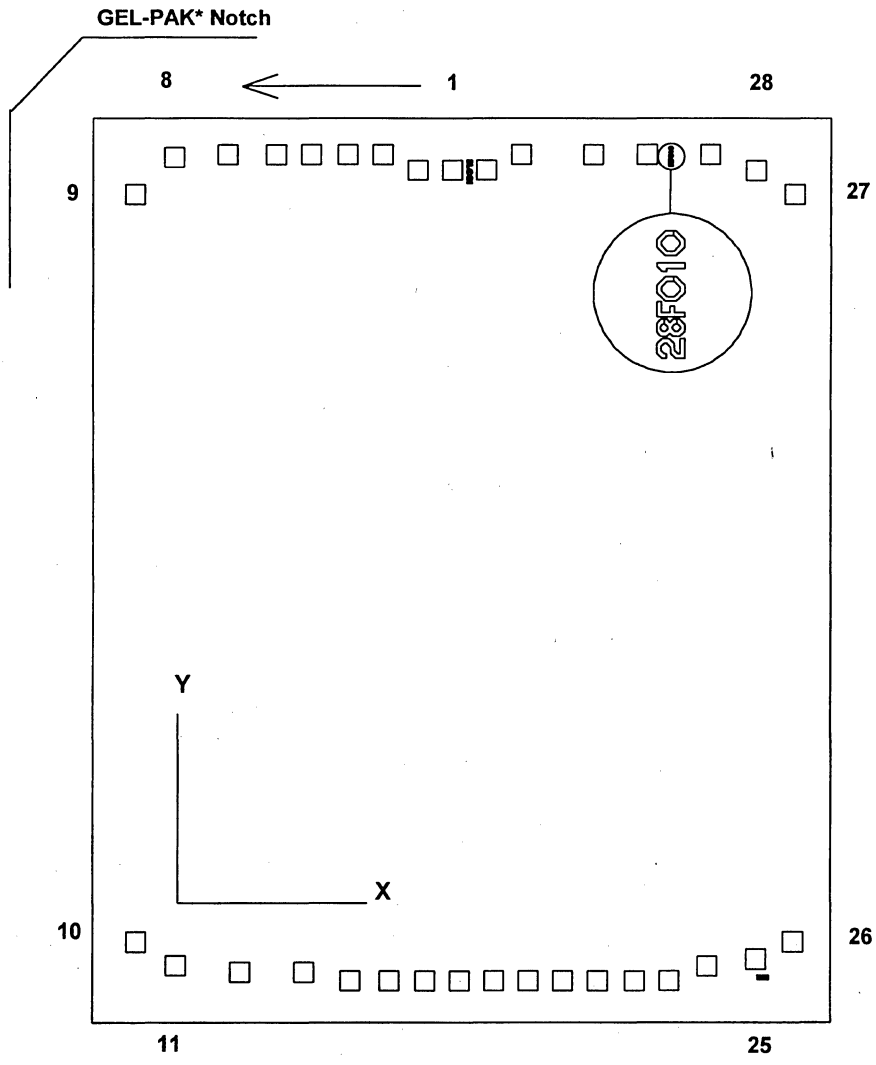
*Other brands and names are the property of their respective owners.

October 1994
Order Number: 271294-002



28F010 1024K (128K x 8) Flash Memory Die Photo

1.0 DIE SPECIFICATIONS



NOTE:
This drawing and the photo on the opposite page indicate the actual orientation of the die in the GEL-PAK* (shipping container). The 45° notches shown on the plot and the photo indicate the location of the GEL-PAK notch.

Figure 1. 28F010 1024K (128K x 8) Flash Memory Die/Bond Pad Layout

1.1 Pad Description

Table 1. 28F010 1024K (128K x 8) Flash Memory Bond Pad Center Data

Pad	Signal	Pad Center			
		(Mils = 0.001 in.)		(Microns)	
		X	Y	X	Y
001	V_{CC}	−2.1	84.2	−53	2138
002	V_{PP}	−9.4	84.2	−238	2138
003	A16	−16.7	87.4	−425	2219
004	A15	−24.2	87.4	−615	2219
005	A12	−31.9	87.4	−810	2219
006	A7	−39.3	87.4	−998	2219
007	A6	−49.5	87.4	−1257	2219
008	A5	−60.8	86.8	−1545	2204
009	A4	−69.1	79.0	−1756	2007
010	A3	−68.9	−78.1	−1750	−1983
011	A2	−60.5	−83.0	−1536	−2109
012	A1	−46.9	−84.4	−1190	−2144
013	A0	−33.3	−84.4	−845	−2144
014	DQ0	−23.6	−86.2	−601	−2190
015	DQ1	−15.3	−86.2	−389	−2190
016	DQ2	−7.8	−86.2	−198	−2190
017	V_{SS}	−0.4	−86.2	−11	−2190
018	V_{SS}	6.8	−86.2	174	−2190
019	DQ3	14.1	−86.2	359	−2190
020	DQ4	21.5	−86.2	545	−2190
021	DQ5	28.9	−86.2	735	−2190
022	DQ6	36.7	−86.2	932	−2190
023	DQ7	44.0	−86.2	1117	−2190
024	CE#	52.1	−83.2	1322	−2114
025	A10	62.2	−81.7	1579	−2076
026	OE#	70.1	−78.1	1781	−1983
027	A11	70.5	79.0	1790	2007
028	A9	62.2	84.0	1580	2133
029	A8	52.6	87.4	1337	2219
030	A13	39.3	87.4	998	2219

Table 1. 28F010 1024K (128K x 8) Flash Memory Bond Pad Center Data (Continued)

Pad	Signal	Pad Center			
		(Mils = 0.001 in.)		(Microns)	
		X	Y	X	Y
031	A14	28.0	87.4	710	2219
032	WE#	12.5	87.4	318	2219
033	V_{CC}	5.2	84.2	132	2138

Notes:
1. **The symbol "#" is used at the end of the signal name to denote an active low signal.**
2. **X-Y pad coordinates represent bond pad centers and are relative to center of die.**

2.0 INTEL DIE PRODUCTS PROCESSING

TEST PROCEDURE

Intel has instituted full-speed functional testing at the die level for all SmartDie products. This level of testing is ordinarily performed only after assembly into a package. Each die is tested to the same electrical limits as the equivalent packaged unit.

WAFER PROBE

Wafer probing is performed on every wafer produced in an Intel Fab. The process consists of specific electrical tests and device-specific functionality tests.

At the wafer level, built-in test structures are probed to verify that device electrical characteristics are in control and meet specifications. Measurements are made of transistor threshold voltages and current characteristics; poly and contact resistance; gate oxide and junction integrity; and specific parameters critical to the particular technology and device type. Wafer-to-wafer, across-the-wafer run-to-run variation and conformance to spec limits are checked.

The actual devices on each wafer are then probed for both functionality and performance to specifications. Additional reliability tests are also included in the probe steps.

WAFER SAW

Probed wafers are transferred to Intel's assembly sites to be sawed. The saw cuts completely through the wafer.

DIE INSPECTION

Upon completion of the wafer saw, the die are moved to pick and place equipment that removes reject die. The remaining die are submitted to the same visual inspection as standard packaged product. The compliant die are then transferred to GEL-PAKs for shipment.

PACKING PROCEDURE

Intel will ship all Intel die products in GEL-PAKs. GEL-PAKs eliminate the die edge damage usually associated with die cavity plates or chip trays.

The backside of each die adheres to the gel membrane in the GEL-PAK, eliminating the risk of damage to the active die surface. A simple vacuum release mechanism allows for pick and place removal at the customer's site.

Only die from the same wafer lot are packaged together in a GEL-PAK, and all die are placed in the GEL-PAKs with a consistent orientation. The GEL-PAKs are then sealed and labeled with the following information:

- Intel Die Products
- Intel Part Number
- Spec
- Customer Part Number (if applicable)
- Fab Lot Number
- Quantity
- Assembly Lot Traveler Number
- Seal Date
- ROM Code (if applicable)

NOTE:
GEL-PAKs require a Vacuum Release Station. Contact Vichem Corporation for more information.

INSPECTION STEPS

Multiple inspection steps are performed during the die fabrication and packing flow. These steps are performed according to the same specifications and criteria established for Intel's standard packaged product. Specific inspection steps include a wafer saw visual as well as a final die visual just before die are sealed in moisture barrier bags.

STORAGE REQUIREMENTS

Intel die products will be shipped in GEL-PAKs and sealed in a moisture-barrier anti-static bag with a desiccant. No special storage procedures are required while the bag is still unopened. Once opened, the GEL-PAK should be stored in a dry, inert atmosphere to prevent bond pad corrosion.

ESD

Components are ESD sensitive.

3.0 SPECIFICATIONS

Specifications within this document are specific to a particular die revision and are subject to change without notice. Verify with your local Intel Sales Office that you have the latest data before finalizing a design.

3.1 28F010 1024K (128K x 8) Flash Memory Handling Requirements

There are two key areas of concern for the 28F010. The first is our recommendation to avoid exposing Flash devices to ultraviolet light. Erasing the device under a UV light erases not only the bits in the array but also the device-specific control information stored in the chips. The second area of concern is in the exposure of the 28F010 to temperatures above 475°C for more than 10 minutes. Exposure above this time/temperature envelope may cause damage to the device reference cells.

3.2 28F010 1024K (128K x 8) Flash Memory Physical Specifications

Substrate Bias Condition: Float (self-biasing to V_{SS}). Alternative is to drive V_{SS}.

Post-saw die dimensions:
Mils: X = 155 ±0.5, Y = 189 ±0.5
See associated Die/Bond Pad Layout for X, Y orientation.

Die Backside: Polished bare silicon.

Pad Passivation Opening Size:
Mils: 4.1 x 4.1
Microns: 105.0 x 105.0

Die Thickness: 17 ± 1 mils

Pad Pitch:
Pads are not all evenly pitched. Minimum pitch is 185 microns (7.3 mils).

Bond Pad Metalization (outermost layer listed first):
0.9 micron Aluminum (0.5% Copper)
0.1 micron Titanium

Die Revision: A

Pads per Die: 33

Intel Fabrication Process: ETOX™ III (minimum feature size: 0.8 micron).

Passivation: (from top surface)
2.3 microns B-Pyrox
1.1 microns Oxynitride

3.3 DC Specifications

ABSOLUTE MAXIMUM RATINGS*

GEL-PAK Storage Temperature ..... 0°C to +70°C
For Junction Temperature Under Bias, Supply Voltage with Respect to V_{SS} and Voltage on Other Pads reference the 28F010 1024K (128K x 8) CMOS Flash Memory Data Sheet, #290207.

OPERATING CONDITIONS*

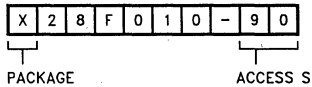
V_{CC} (Digital Supply Voltage) ..... 4.5V to 5.5V
T_J (Junction Temperature Under Bias) ..... 0°C to 80°C

NOTICE: This document contains prelminary information on new products in production. It is valid for the new devices indicated in the revision history. This specification is subject to change without notice.

****WARNING:** Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

10

4.0 DEVICE NOMENCLATURE



271294-3

VALID COMBINATIONS:
X28F400BX-T80

5.0 ADDITIONAL INFORMATION

Title	Order No.
28F010 1-Mbit Flash Memory	290207
ER-20, ETOX™ II Flash Memory Technology	294005
ER-24, Intel Flash Memory	294008
RR-60, ETOX™ II Flash Memory Reliability Data Summary	293002
AP-316, Using Flash Memory for In-System Reprogrammable Nonvolatile Storage	292046
AP-325, Guide to Flash Memory Reprogramming	292059

6.0 REVISION HISTORY

Rev	Date	Description
001	3/94	Original Version.
002	6/94	Replace Figure 1 to add GEL-PAK orientation information. General re-write of Section 2.0.

28F020 2048K (256K x 8) FLASH MEMORY

SmartDie Product Specification

- **Flash Electrical Chip Erase**
  - **2 Second Typical Chip Erase**
- **Quick-Pulse Programming Algorithm**
  - **10 μs Typical Byte Program**
  - **4 Second Chip Program**
- **100K Erase/Program Cycles Typical**
- **12.0V ±5% V_{PP}**
- **High-Performance Read**
  - **90 ns Access Time**
- **CMOS Low Power Consumption**
  - **10 mA Typical Active Current**
  - **50 μA Typical Standby Current**
  - **0W Data Retention Power**
- **Integrated Program/Erase Stop Timer**
- **Command Register Architecture for Microprocessor/Microcontroller Compatible Write Interface**
- **Noise Immunity Features**
  - **± 10% V_{CC} Tolerance**
  - **Maximum Latch-Up Immunity through EPI Processing**
- **ETOX™ III Nonvolatile Flash Technology**
  - **EPROM-Compatible Process Base**
  - **High-Volume Manufacturing Experience**
- **Intel SmartDie Product**
  - **Full AC/DC Testing at Die Level**
  - **0°C to +80°C (Junction) Temperature Range**
  - **Available in 90 ns Access Time Only**

10

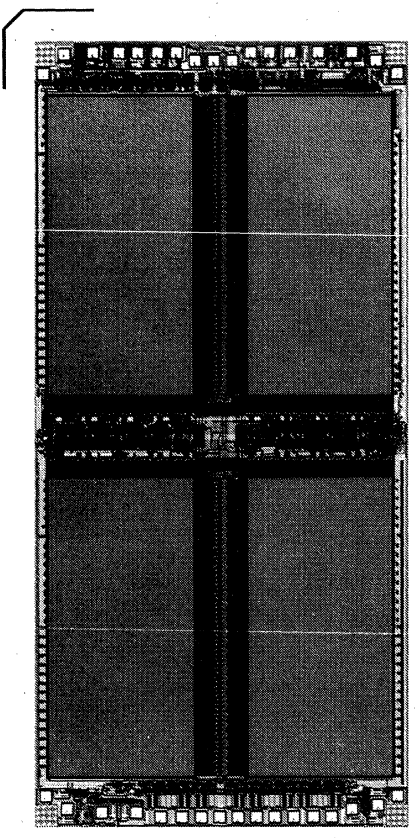
NOTICE: This document contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

REFERENCE INFORMATION: The information in this document is provided as a supplement to the Standard Package Data Sheet on a specific product. Please reference the Standard Package Data Sheet/Book (Order No. 290245) for additional product information and specifications not found in this document.

---

*Other brands and names are the property of their respective owners.

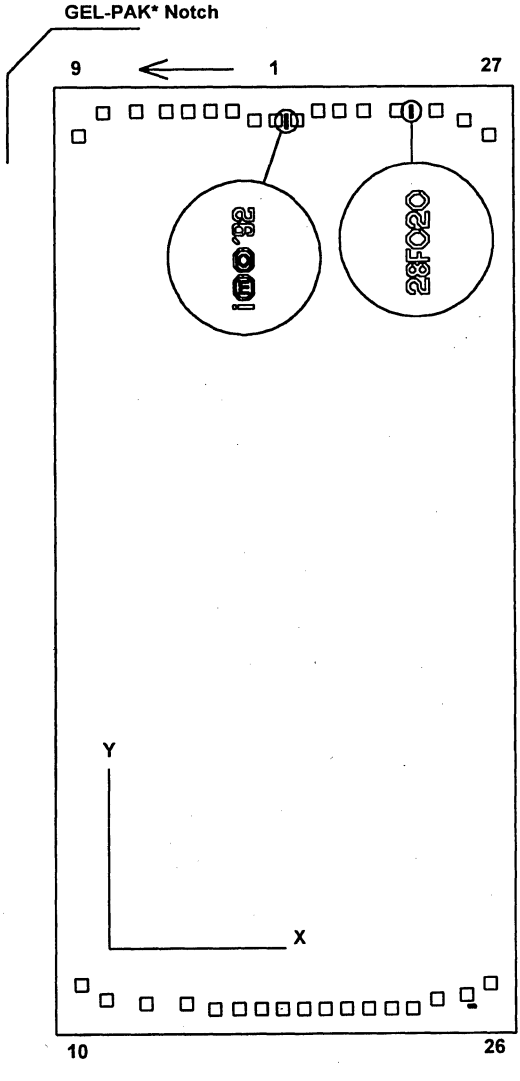
October 1994
Order Number: 271295-002



271295-1

28F020 2048K (256K x 8) Flash Memory Die Photo

1.0 DIE SPECIFICATIONS



NOTE:
This drawing and the photo on the preceding page indicate the actual orientation of the die in the GEL-PAK* (shipping container). The 45° notches shown on the plot and the photo indicate the location of the GEL-PAK notch.

Figure 1. 28F020 2048K (256K x 8) Flash Memory Die/Bond Pad Layout

1.1 Pad Description

Table 1. 28F020 2048K (256K x 8) Flash Memory Bond Pad Center Data

Pad	Signal	Pad Center			
		(Mils = 0.001 in.)		(Microns)	
		X	Y	X	Y
001	V_{CC}	−2.7	147.0	−68	3735
002	V_{PP}	−9.9	147.0	−253	3735
003	A16	−17.3	150.3	−439	3817
004	A15	−24.8	150.3	−629	3817
005	A12	−32.5	150.3	−825	3817
006	A7	−39.9	150.3	−1012	3817
007	A6	−50.1	150.3	−1272	3817
008	A5	−61.4	149.7	−1560	3801
009	A4	−69.7	141.9	−1770	3604
010	A3	−69.5	−142.1	−1764	−3610
011	A2	−61.0	−147.1	−1550	−3736
012	A1	−47.4	−148.5	−1205	−3771
013	A0	−33.9	−148.5	−860	−3771
014	DQ0	−24.2	−150.3	−615	−3817
015	DQ1	−15.9	−150.3	−404	−3817
016	DQ2	−8.4	−150.3	−212	−3817
017	V_{SS}	−1.0	−150.3	−26	−3817
018	V_{SS}	6.3	−150.3	159	−3817
019	DQ3	13.5	−150.3	344	−3817
020	DQ4	20.9	−150.3	530	−3817
021	DQ5	28.4	−150.3	720	−3817
022	DQ6	36.1	−150.3	917	−3817
023	DQ7	43.4	−150.3	1102	−3817
024	CE#	51.5	−147.3	1308	−3741
025	A10	61.6	−145.8	1565	−3703
026	OE#	69.5	−142.1	1766	−3610
027	A11	69.9	141.9	1775	3604
028	A9	61.6	146.9	1565	3730
029	A8	52.1	150.3	1322	3817
030	A13	38.7	150.3	983	3817

1.1 Pad Description

Table 1. 28F020 2048K (256K x 8) Flash Memory Bond Pad Center Data

Pad	Signal	Pad Center			
		(Mils = 0.001 in.)		(Microns)	
		X	Y	X	Y
031	A14	27.4	150.3	695	3817
032	A17	19.3	150.3	490	3817
033	WE#	11.9	150.3	303	3817
034	V_{CC}	4.6	147.0	117	3735

Notes:
1. The symbol "#" is used at the end of the signal name to denote an active low signal.
2. X-Y pad coordinates represent bond pad centers and are relative to center of die.

2.0 INTEL DIE PRODUCTS PROCESSING

TEST PROCEDURE

Intel has instituted full-speed functional testing at the die level for all SmartDie products. This level of testing is ordinarily performed only after assembly into a package. Each die is tested to the same electrical limits as the equivalent packaged unit.

WAFER PROBE

Wafer probing is performed on every wafer produced in an Intel Fab. The process consists of specific electrical tests and device-specific functionality tests.

At the wafer level, built-in test structures are probed to verify that device electrical characteristics are in control and meet specifications. Measurements are made of transistor threshold voltages and current characteristics; poly and contact resistance; gate oxide and junction integrity; and specific parameters critical to the particular technology and device type. Wafer-to-wafer, across-the-wafer run-to-run variation and conformance to spec limits are checked.

The actual devices on each wafer are then probed for both functionality and performance to specifications. Additional reliability tests are also included in the probe steps.

WAFER SAW

Probed wafers are transferred to Intel's assembly sites to be sawed. The saw cuts completely through the wafer.

DIE INSPECTION

Upon completion of the wafer saw, the die are moved to pick and place equipment that removes reject die. The remaining die are submitted to the same visual inspection as our standard packaged product. The compliant die are then transferred to GEL-PAKs for shipment.

PACKING PROCEDURE

Intel will ship all Intel die products in GEL-PAKs. GEL-PAKs eliminate the die edge damage usually associated with die cavity plates or chip trays.

The backside of each die adheres to the gel membrane in the GEL-PAK, eliminating the risk of damage to the active die surface. A simple vacuum release mechanism allows for pick and place removal at the customer's site.

Only die from the same wafer lot are packaged together in a GEL-PAK, and all die are placed in the GEL-PAKs with a consistent orientation. The GEL-PAKs are then sealed and labeled with the following information:

- Intel Die Products
- Intel Part Number
- Spec
- Customer Part Number (if applicable)
- Fab Lot Number
- Quantity
- Assembly Lot Traveler Number
- Seal Date
- ROM Code (if applicable)

NOTE:
GEL-PAKs require a Vacuum Release Station. Contact Vichem Corporation for more information.

INSPECTION STEPS

Multiple inspection steps are performed during the die fabrication and packing flow. These steps are performed according to the same specifications and criteria established for Intel's standard packaged product. Specific inspection steps include a wafer saw visual as well as a final die visual just before die are sealed in moisture barrier bags.

STORAGE REQUIREMENTS

Intel die products will be shipped in GEL-PAKs and sealed in a moisture-barrier anti-static bag with a desiccant. No special storage procedures are required while the bag is still unopened. Once opened, the GEL-PAK should be stored in a dry, inert atmosphere to prevent bond pad corrosion.

ESD

Components are ESD sensitive.

3.0 SPECIFICATIONS

Specifications within this document are specific to a particular die revision and are subject to change without notice. Verify with your local Intel Sales Office that you have the latest data before finalizing a design.

3.1 28F020 2048K (256K x 8) Flash Memory Handling Requirements

There are two key areas of concern for the 28F020 Flash Memory product. The first is our recommendation to avoid exposing Flash devices to ultraviolet light. Erasing the device under a UV light erases not only the bits in the array but also the device-specific control information stored in the chips. The second area of concern is in the exposure of the 28F020 Flash Memory product to temperatures above 475°C for more than 10 minutes. Exposure above this time/temperature envelope may cause damage to the device reference cells.

3.2 28F020 2048K (256K x 8) Flash Memory Physical Specifications

Substrate Bias Condition: Self-biasing to V_{SS}. Alternative is to drive V_{SS}.

3.3 DC Specifications

ABSOLUTE MAXIMUM RATINGS*

GEL-PAK Storage Temperature ..... 0°C to +70°C

For Junction Temperature under Bias, Supply Voltage with Respect to V_{SS} and Voltage on Other Pads reference the 28F020 2048K (256K x 8) CMOS Flash Memory Data Sheet, Order No. 290245.

OPERATING CONDITIONS*

Digital Supply Voltage (V_{CC}) ..... 4.5V to 5.5V

Junction Temperature under Bias (T_J) ..... 0°C to 80°C

Post-saw die dimensions:
Mils: X = 155 ±0.5, Y = 316 ±0.5
See associated Die/Bond Pad Layout for X,Y orientation.

Die Backside: Polished bare silicon.

Pad Passivation Opening Size:
Mils: 4.1 x 4.1
Microns: 105.0 x 105.0

Die Thickness: 17 ± 1 mils

Pad Pitch:
Pads are not all evenly pitched. Minimum pitch is 185 microns (7.3 mils).

Bond Pad Metalization (outermost layer listed first)
0.9 microns Aluminum (0.5% copper)
0.1 micron Titanium

Die Revision: B

Pads per Die: 34

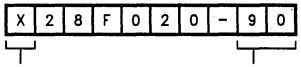
Intel Fabrication Process: ETOX™ III (minimum feature size 0.8 micron).

Passivation: (from top surface)
2.3 microns B-Pyrox
1.1 microns Oxynitride

NOTICE: This document contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

4.0 DEVICE NOMENCLATURE



271295-3

VALID COMBINATIONS:
X28F020-90

5.0 REFERENCE INFORMATION

Title	Order No.
28F020 2-Mbit Flash Memory	290245
ER-20, ETOX™ II Flash Memory Technology	294005
ER-24, Intel Flash Memory	294008
RR-60, ETOX™ II Flash Memory Reliability Data Summary	293002
AP-316, Using Flash Memory for In-System Reprogrammable Nonvolatile Storage	292046
AP-325, Guide to Flash Memory Reprogramming	292059

6.0 REVISION HISTORY

Rev	Date	Description
001	3/94	Original version.
002	6/94	Replace Figure 1 to add GEL-PAK orientation information. General rewrite of Section 2.0.

28F008SA 8-MBIT (1-MBIT x 8) FLASH MEMORY

SmartDie Product Specification

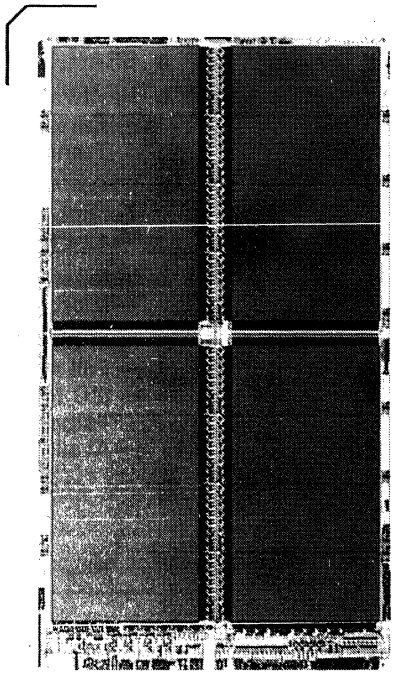
- **High-Density Symmetrically Blocked Architecture**
  - **Sixteen 64-Kbyte Blocks**
- **Extended Cycling Capability**
  - **100K Block Erase Cycles**
  - **1.6M Block Erase Cycles per Chip**
- **Automated Byte Write and Block Erase**
  - **Command User Interface**
  - **Status Register**
- **System Performance Enhancements**
  - **RY/BY# Status Output**
  - **Erase Suspend Capability**
- **Deep Powerdown Mode**
  - **0.2 μA I_{CC} Typical**
- **High-Performance Read**
  - **120 ns Maximum Access Time**
- **SRAM-Compatible Write Interface**
- **Hardware Data Protection Feature**
  - **Erase/Write Lockout During Power Transitions**
- **ETOX™ III Nonvolatile Flash Technology**
  - **12V Byte Write/Block Erase**
- **Independent Software Vendor Support**
  - **Microsoft* Flash File System (FFS)**
- **Intel SmartDie Product**
  - **Full AC/DC Testing at Die Level**
  - **0°C to +80°C (Junction) Temperature Range**
  - **Available in 120 ns Access Time Only**

NOTICE: This data sheet contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

REFERENCE INFORMATION: The information in this document is provided as a supplement to the Standard Package Data Sheet on a specific product. Please reference the Standard Package Data Sheet/Book (Order No. 290429) for additional product information and specifications not found in this document.

*Other brands and names are the property of their respective owners.

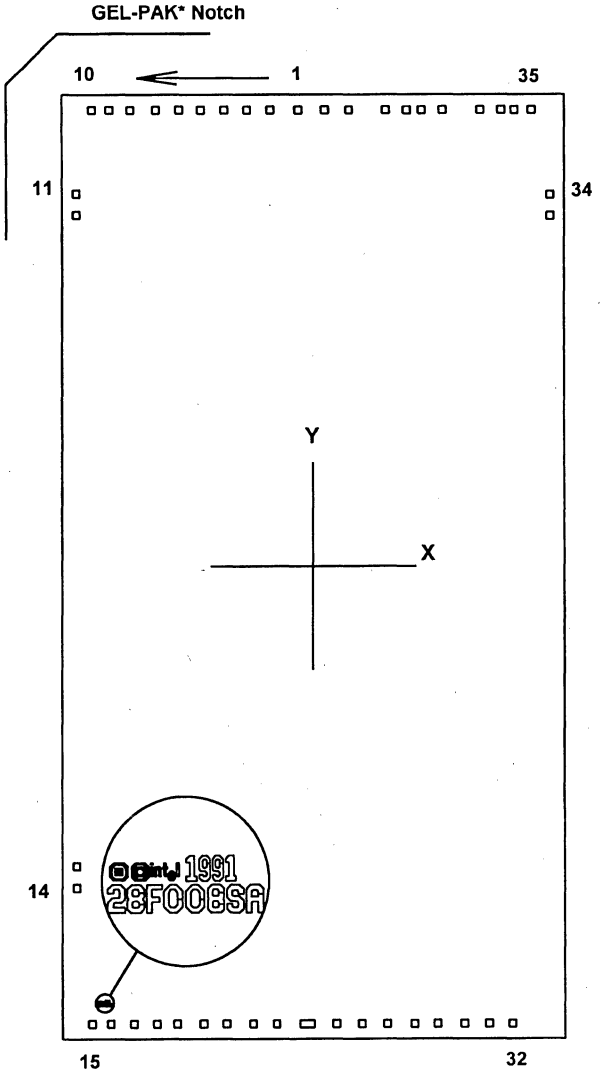
October 1994
Order Number: 271296-002



271296-1

28F008SA 8-Mbit (1-Mbit x 8) Flash Memory Die Photo

1.0 DIE SPECIFICATIONS



NOTE:
This drawing and the photo on the opposite page indicate the actual orientation of the die in the GEL-PAK* (shipping container). The 45 ° notches shown on the plot and the photo indicate the location of the GEL-PAK notch.

Figure 1. 28F008SA 8-Mbit (1-Mbit x 8) Flash Memory Die/Bond Pad Layout

1.1 Pad Description

Table 1. 28F008SA 8-Mbit (1-Mbit x 8) Flash Memory Bond Pad Center Data

Pad	Signal	Pad Center			
		(Inches)		(Microns)	
		X	Y	X	Y
001	V_{PP}	−0.0084	0.2652	−212	6736
002	PWD#	−0.0248	0.2652	−630	6736
003	A11	−0.0387	0.2652	−982	6736
004	A10	−0.0523	0.2652	−1328	6736
005	A9	−0.0660	0.2652	−1676	6736
006	A8	−0.0789	0.2652	−2004	6736
007	A7	−0.0927	0.2652	−2354	6736
008	A6	−0.1075	0.2652	−2731	6736
009	A5	−0.1199	0.2652	−3045	6736
010	A4	−0.1301	0.2652	−3306	6736
011	N.C.	−0.1392	0.2165	−3535	5498
012	N.C.	−0.1392	0.2041	−3535	5184
013	N.C.	−0.1392	−0.1740	−3535	−4420
014	N.C.	−0.1392	−0.1864	−3535	−4734
015	A3	−0.1301	−0.2652	−3305	−6736
016	A2	−0.1191	−0.2652	−3025	−6736
017	A1	−0.1055	−0.2652	−2681	−6736
018	A0	−0.0922	−0.2652	−2341	−6736
019	DQ0	−0.0801	−0.2652	−2036	−6736
020	DQ1	−0.0644	−0.2652	−1637	−6736
021	DQ2	−0.0510	−0.2652	−1294	−6736
022	DQ3	−0.0353	−0.2652	−896	−6736
023	V_{SS}	−0.0215	−0.2652	−545	−6736
024	V_{SS}(4)	−0.0037	−0.2652	−95	−6736
025	V_{CC}	0.0135	−0.2652	342	−6736
026	DQ4	0.0280	−0.2652	712	−6736
027	DQ5	0.0431	−0.2652	1094	−6736
028	DQ6	0.0590	−0.2652	1499	−6736
029	DQ7	0.0725	−0.2652	1841	−6736
030	RY/BY#	0.0883	−0.2652	2242	−6736
031	OE#	0.1028	−0.2652	2611	−6736
032	WE#	0.1164	−0.2652	2957	−6736

Table 1. 28F008SA 8-Mbit (1-Mbit x 8) Flash Memory Bond Pad Center Data (Continued)

Pad	Signal	Pad Center			
		(Inches)		(Microns)	
		X	Y	X	Y
033	N.C.	0.1392	0.2032	3535	5162
034	N.C.	0.1392	0.2156	3535	5476
035	A19	0.1282	0.2652	3258	6736
036	A18	0.1180	0.2652	2996	6736
037	A17	0.1103	0.2652	2801	6736
038	A16	0.0979	0.2652	2488	6736
039	A15	0.0758	0.2652	1925	6736
040	A14	0.0634	0.2652	1611	6736
041	A13	0.0547	0.2652	1389	6736
042	A12	0.0423	0.2652	1076	6736
043	CE#	0.0210	0.2652	533	6736
044	V_{CC}	0.0068	0.2652	173	6736

Notes:
1. N.C. signifies no connect. These pads must not be connected.
2. The symbol "#" is used at the end of the signal name to denote an active low signal.
3. X-Y pad coordinates represent bond pad centers and are relative to center of die.
4. Double wide bond pad.

2.0 INTEL DIE PRODUCTS PROCESSING

TEST PROCEDURE

Intel has instituted full-speed functional testing at the die level for all SmartDie products. This level of testing is ordinarily performed only after assembly into a package. Each die is tested to the same electrical limits as the equivalent packaged unit.

WAFER PROBE

Wafer probing is performed on every wafer produced in an Intel Fab. The process consists of specific electrical tests and device-specific functionality tests.

At the wafer level, built-in test structures are probed to verify that device electrical characteristics are in control and meet specifications. Measurements are made of transistor threshold voltages and current characteristics; poly and contact resistance; gate oxide and junction integrity; and specific parameters critical to the particular technology and device type. Wafer-to-wafer, across-the-wafer run-to-run variation and conformance to spec limits are checked.

The actual devices on each wafer are then probed for both functionality and performance to specifications. Additional reliability tests are also included in the probe steps.

WAFER SAW

Probed wafers are transferred to Intel's assembly sites to be sawed. The saw cuts completely through the wafer.

DIE INSPECTION

Upon completion of the wafer saw, the die are moved to pick and place equipment that removes reject die. The remaining die are submitted to the same visual inspection as standard packaged product. The compliant die are then transferred to GEL-PAKs for shipment.

PACKING PROCEDURE

Intel will ship all Intel die products in GEL-PAKs. GEL-PAKs eliminate the die edge damage usually associated with die cavity plates or chip trays.

The backside of each die adheres to the gel membrane in the GEL-PAK, eliminating the risk of damage to the active die surface. A simple vacuum release mechanism allows for pick and place removal at the customer's site.

Only die from the same wafer lot are packaged together in a GEL-PAK, and all die are placed in the GEL-PAKs with a consistent orientation. The GEL-PAKs are then sealed and labeled with the following information:

- Intel Die Products
- Intel Part Number
- Spec
- Customer Part Number (if applicable)
- Fab Lot Number
- Quantity
- Assembly Lot Traveler Number
- Seal Date
- ROM Code (if applicable)

NOTE:
GEL-PAKs require a Vacuum Release Station. Contact Vichem Corporation for more information.

INSPECTION STEPS

Multiple inspection steps are performed during the die fabrication and packing flow. These steps are performed according to the same specifications and criteria established for Intel's standard packaged product. Specific inspection steps include a wafer saw visual as well as a final die visual just before die are sealed in moisture barrier bags.

STORAGE REQUIREMENTS

Intel die products will be shipped in GEL-PAKs and sealed in a moisture-barrier anti-static bag with a desiccant. No special storage procedures are required while the bag is still unopened. Once opened, the GEL-PAK should be stored in a dry, inert atmosphere to prevent bond pad corrosion.

ESD

Components are ESD sensitive.

3.0 SPECIFICATIONS

Specifications within this document are specific to a particular die revision and are subject to change without notice. Verify with your local Intel Sales Office that you have the latest data before finalizing a design.

3.1 28F008SA 8-Mbit (1-Mbit x 8) Flash Memory Handling Requirements

There are two key areas of concern for the 28F008SA. The first is our recommendation to avoid exposing Flash devices to ultraviolet light. Erasing the device under a UV light erases not only the bits in the array but also the device-specific control information stored in the chips. The second area of concern is in the exposure of the 28F008SA to temperatures above 350°C for more than 10 minutes. Exposure above this time/temperature envelope may cause damage to the device reference cells.

3.2 28F008SA 8-Mbit (1-Mbit x 8) Flash Memory Physical Specifications

Substrate Bias Condition: Isolate. Chip self-biases (to V_{SS}). Alternative is to drive V_{SS}.

Post-saw die dimensions:
Mils: X = 294 ±0.5, Y = 547 ±0.5
See associated Die/Bond Pad Layout for X,Y orientation.

Die Backside: Polished bare silicon.

Pad Passivation Opening Size:
Mils: 4.1 x 4.1
Microns: 105 x 105

Die Thickness: 17 ± 1 mils

Pad Pitch:
Pads are not all evenly pitched. Minimum pitch is 7.7 mils between pads 36 and 37.

Bond Pad Metalization (outermost layer listed first):
0.9 microns Aluminum (0.5% Copper)
0.1 micron Titanium

Die Revision: A

Pads per Die: 44

Intel Fabrication Process: ETOX™ III (minimum feature size: 0.8 micron)

Passivation (outermost layer listed first):
2.3 microns B-Pyrox
1.1 microns Oxynitride

3.3 DC Specifications

ABSOLUTE MAXIMUM RATINGS*

GEL-PAK Storage Temperature .....0°C to +70°C
For Junction Temperature Under Bias, Supply Voltage with Respect to V_{SS} and Voltage on Other Pads reference the 28F008SA 8-Mbit (1-Mbit x 8) Flash Memory Data Sheet, Order No. 290429.

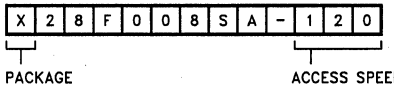
OPERATING CONDITIONS*

V_{CC} (Digital Supply Voltage) .....4.5V to 5.5V
T_J (Junction Temperature Under Bias).....0°C to +80°C

NOTICE: This document contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

****WARNING:** Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

4.0 DEVICE NOMENCLATURE



VALID COMBINATIONS:
X28F008SA-120

5.0 REFERENCE INFORMATION

Title	Order No.
28F008SA 8-Mbit FlashFile Memory Data Sheet	290429
AP-359, 28F008SA Hardware Interfacing	292094
AP-360, 25F008SA Software Drivers	292095
AP-364, 28F008SA Automation and Algorithms	292099
ER-27, The Intel 28F008SA Flash Memory	294011
ER-28, ETOX™ III Flash Memory Technology	290412

6.0 REVISION HISTORY

Rev	Date	Description
001	3/94	Original version
002	6/94	Replace Figure 1 to add GEL-PAK orientation information. General re-write of Section 2.0.

28F400BX-T
4-MBIT (256K x 16, 512K x 8)
BOOT BLOCK FLASH MEMORY

SmartDie Product Specification

- x8/x16 Input/Output Architecture
  - 28F400BX-T
  - For High-Performance and High-Integration 16-Bit and 32-Bit CPUs
- Optimized High-Density Blocked Architecture
  - One 16 KByte Protected Boot Block
  - Two 8 KByte Parameter Blocks
  - One 96 KByte Main Block
  - Three 128 KByte Main Blocks
  - Top Boot Location
- Write Protection for Boot Block
- Hardware Data Protection Feature
  - Erase/Write Lockout During Power Transitions
- Extended Cycling Capability
  - 100K Block Erase Cycles
- Automated Word/Byte Write and Block Erase
  - Command User Interface
  - Status Registers
  - Erase Suspend Capability
- SRAM-Compatible Write Interface
- Automatic Power Savings Feature
  - 1 mA I_{CC} Active Current in Static Operation
- Very High-Performance Read
  - 80 ns Maximum Access Time
  - 40 ns Maximum Output Enable Time
- Low Power Consumption
  - 20 mA Typical x8 Active Read Current
  - 25 mA Typical x16 Active Read Current
- Reset/Deep Power-Down Input
  - 0.2 μA I_{CC} Typical
  - Acts as Reset for Boot Operations
- 12V Word/Byte Write and Block Erase
  - V_{PP} = 12V $\pm$5%
- ETOX™ III Nonvolatile Flash Technology
  - 5V Read
- Intel SmartDie Product
  - Full AC/DC Testing at Die Level
  - 0°C to +80°C (Junction) Temperature Range
  - Available in 80 ns Access Time, Top Boot Version Only

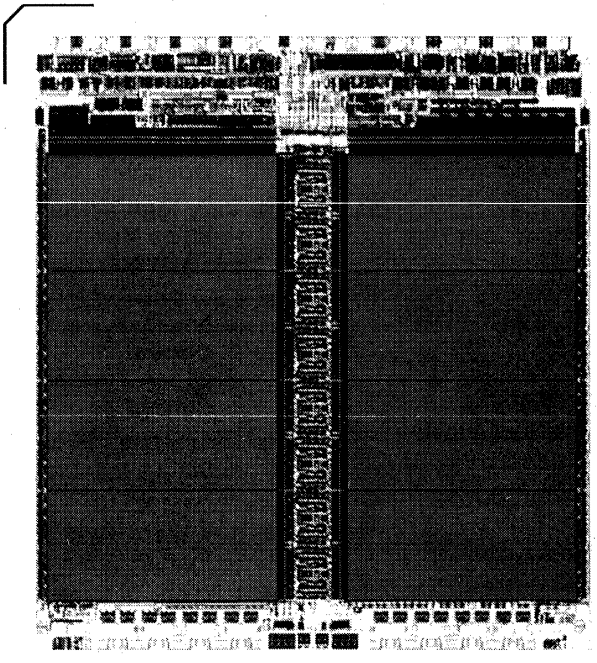
NOTICE: This document contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

REFERENCE INFORMATION: The information in this document is provided as a supplement to the Standard Package Data Sheet on a specific product. Please reference the Standard Package Data Sheet/Book (Order No. 290451) for additional product information and specifications not found in this document.

---

*Other brands and names are the property of their respective owners.

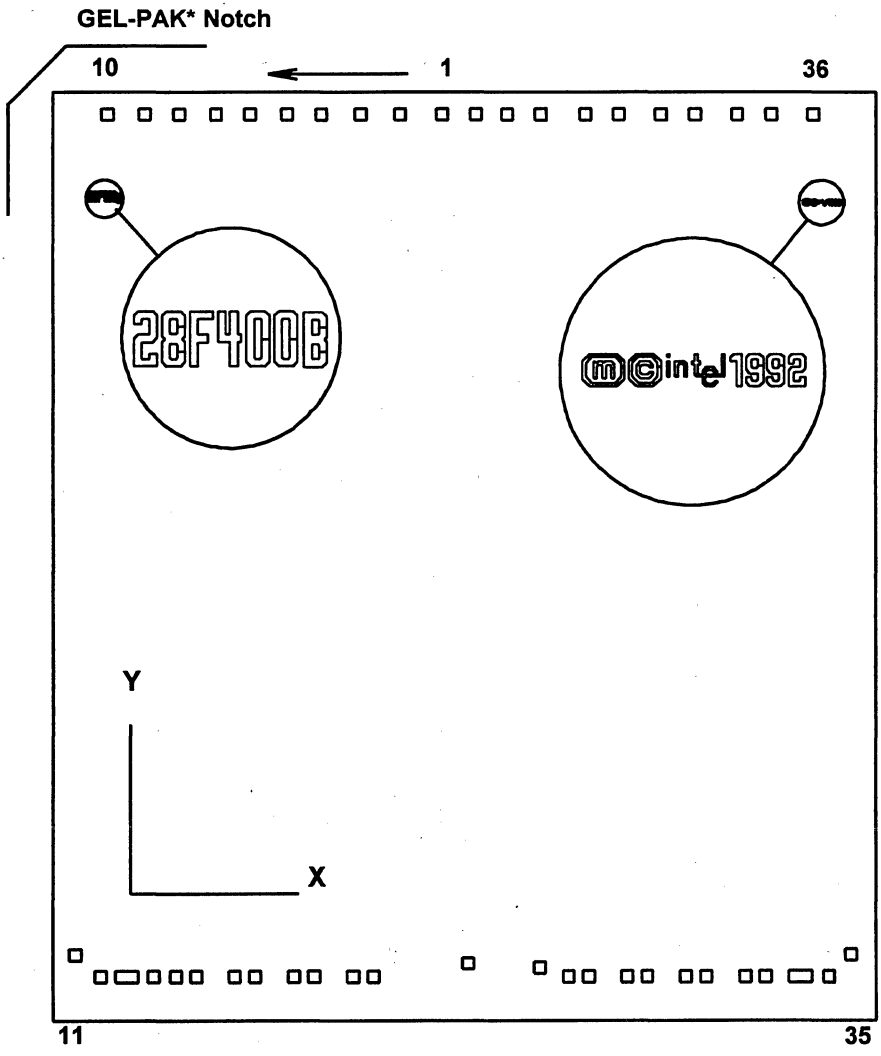
October 1994
Order Number: 271302-002



271302-1

28F400BX-T 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Die Photo

1.0 DIE SPECIFICATIONS



NOTE:
This drawing and the photo on the opposite page indicate the actual orientation of the die in the GEL-PAK* (shipping container). The 45° notches shown on the plot and the photo indicate the location of the GEL-PAK notch.

Figure 1. 28F400BX-T 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Die/Bond Pad Layout

1.1 Pad Description

Table 1. 28F400BX-T 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Bond Pad Center Data

Pad	Signal	Pad Center			
		(Inches)		(Microns)	
		X	Y	X	Y
001	V_{PP}	−0.0086	0.1575	−218	4001
002	N.C.	−0.0236	0.1575	−599	4001
003	A17	−0.0376	0.1575	−955	4001
004	A7	−0.0518	0.1575	−1316	4001
005	A6	−0.0640	0.1575	−1626	4001
006	A5	−0.0773	0.1575	−1963	4001
007	A4	−0.0895	0.1575	−2273	4001
008	A3	−0.1027	0.1575	−2609	4001
009	A2	−0.1149	0.1575	−2918	4001
010	A1	−0.1282	0.1575	−3256	4001
011	A0	−0.1396	−0.1420	−3546	−3607
012	CE#	−0.1307	−0.1497	−3320	−3802
013	V_{SS}[4]	−0.1213	−0.1497	−3081	−3802
014	OE#	−0.1119	−0.1498	−2842	−3805
015	DQ0	−0.1038	−0.1497	−2637	−3802
016	DQ8	−0.0966	−0.1497	−2454	−3802
017	DQ1	−0.0827	−0.1497	−2101	−3802
018	DQ9	−0.0754	−0.1497	−1915	−3802
019	DQ2	−0.0615	−0.1497	−1562	−3802
020	DQ10	−0.0542	−0.1497	−1377	−3802
021	DQ3	−0.0403	−0.1497	−1024	−3802
022	DQ11	−0.0331	−0.1497	−841	−3802
023	V_{CC}	0.0008	−0.1452	20	−3688
024	V_{CC}	0.0265	−0.1467	673	−3726
025	DQ4	0.0370	−0.1497	940	−3802
026	DQ12	0.0442	−0.1497	1123	−3802
027	DQ5	0.0581	−0.1497	1476	−3802
028	DQ13	0.0654	−0.1497	1661	−3802
029	DQ6	0.0793	−0.1497	2014	−3802
030	DQ14	0.0866	−0.1497	2200	−3802
031	DQ7	0.1004	−0.1497	2550	−3802
032	DQ15/A-1	0.1077	−0.1497	2736	−3802

Table 1. 28F400BX-T 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Bond Pad Center Data

Pad	Signal	Pad Center			
		(Inches)		(Microns)	
		X	Y	X	Y
033	V_{SS}(4)	0.1204	−0.1497	3058	−3802
034	BYTE#	0.1307	−0.1497	3320	−3802
035	A16	0.1386	−0.1420	3520	−3607
036	A15	0.1250	0.1575	3175	4001
037	A14	0.1099	0.1575	2791	4001
038	A13	0.0977	0.1575	2482	4001
039	A12	0.0826	0.1575	2098	4001
040	A11	0.0703	0.1575	1786	4001
041	A10	0.0552	0.1575	1402	4001
042	A9	0.0430	0.1575	1092	4001
043	A8	0.0269	0.1575	683	4001
044	WE#	0.0147	0.1575	373	4001
045	PWD#	0.0035	0.1575	89	4001

Notes:
1. **N.C. signifies no connect. These pads must not be connected.**
2. **The symbol "#" is used at the end of the signal name to denote an active low signal.**
3. **X-Y pad coordinates represent bond pad centers and are relative to center of die.**
4. **Double wide bond pad.**

2.0 INTEL DIE PRODUCTS PROCESSING

TEST PROCEDURE

Intel has instituted full-speed functional testing at the die level for all SmartDie products. This level of testing is ordinarily performed only after assembly into a package. Each die is tested to the same electrical limits as the equivalent packaged unit.

WAFER PROBE

Wafer probing is performed on every wafer produced in an Intel Fab. The process consists of specific electrical tests and device-specific functionality tests.

At the wafer level, built-in test structures are probed to verify that device electrical characteristics are in control and meet specifications. Measurements are made of transistor threshold voltages and current characteristics; poly and contact resistance; gate oxide and junction integrity; and specific parameters critical to the particular technology and device type. Wafer-to-wafer, across-the-wafer run-to-run variation and conformance to spec limits are checked.

The actual devices on each wafer are then probed for both functionality and performance to specifications. Additional reliability tests are also included in the probe steps.

WAFER SAW

Probed wafers are transferred to Intel's assembly sites to be sawed. The saw cuts completely through the wafer.

DIE INSPECTION

Upon completion of the wafer saw, the die are moved to pick and place equipment that removes reject die. The remaining die are submitted to the same visual inspection as our standard packaged product. The compliant die are then transferred to GEL-PAKs for shipment.

PACKING PROCEDURE

Intel will ship all Intel die products in GEL-PAKs. GEL-PAKs eliminate the die edge damage usually associated with die cavity plates or chip trays.

The backside of each die adheres to the gel membrane in the GEL-PAK, eliminating the risk of damage to the active die surface. A simple vacuum release mechanism allows for pick and place removal at the customer's site.

Only die from the same wafer lot are packaged together in a GEL-PAK, and all die are placed in the GEL-PAKs with a consistent orientation. The GEL-PAKs are then sealed and labeled with the following information:

- Intel Die Sales
- Intel Part Number
- Spec
- Customer Part Number (if applicable)
- Fab Lot Number
- Quantity
- Assembly Lot Traveler Number
- Seal Date
- ROM Code (if applicable)

NOTE:
GEL-PAKs require a Vacuum Release Station. Contact Vichem Corporation for more information.

INSPECTION STEPS

Multiple inspection steps are performed during the die fabrication and packing flow. These steps are performed according to the same specifications and criteria established for Intel's standard packaged product. Specific inspection steps include a wafer saw visual as well as a final die visual just before die are sealed in moisture barrier bags.

STORAGE REQUIREMENTS

Intel die products will be shipped in GEL-PAKs and sealed in a moisture-barrier anti-static bag with a desiccant. No special storage procedures are required while the bag is still unopened. Once opened, the GEL-PAK should be stored in a dry, inert atmosphere to prevent bond pad corrosion.

ESD

Components are ESD sensitive.

3.0 SPECIFICATIONS

Specifications within this document are specific to a particular die revision and are subject to change without notice. Verify with your local Intel Sales Office that you have the latest data before finalizing a design.

3.1 28F400BX-T 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Handling Requirements

There are two key areas of concern for the 28F400BX-T. The first is our recommendation to avoid exposing Flash devices to ultraviolet light. Erasing the device under a UV light erases not only the bits in the array but also the device-specific control information stored in the chips. The second area of concern is in the exposure of the 28F400BX-T to temperatures above 475°C for more than 10 minutes. Exposure above this time/temperature envelope may cause damage to the device reference cells.

3.2 28F400BX-T 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Physical Specifications

Substrate Bias Condition: Float (self-biasing to V_{SS}). Alternative is to drive V_{SS}.

3.3 DC Specifications

ABSOLUTE MAXIMUM RATINGS*

GEL-PAK Storage Temperature ..... 0°C to +70°C

For Junction Temperature Under Bias, Supply Voltage with Respect to V_{SS} and Voltage on Other Pads reference the 28F400BX-T/B, 28F004BX-T/B 4-Mbit (256K x 16, 512K x 8) Boot Block Flash Memory Family Data Sheet, Order No. 290451.

OPERATING CONDITIONS*

V_{CC} (Digital Supply Voltage) ..... 4.5V to 5.5V

T_J (Junction Temperature Under Bias) ..... 0°C to +80°C

Post-saw Die Dimensions:
Mils: X = 294 ±0.5, Y = 330 ±0.5
See associated Die/Bond Pad Layout for X,Y orientation.

Die Backside: Polished bare silicon.

Pad Passivation Opening Size:
Mils: 4.1 x 4.1 (single pads),
8.2 x 4.1 (double-wide pads)
Microns: 105 x 105 (single pads),
210 x 105 (double-wide pads)

Die Thickness: 17 ± 1 mils

Pad Pitch:
Pads are not all evenly pitched. Minimum pitch is 185 microns (7.3 mils).

Bond Pad Metalization (outermost layer listed first):
0.9 microns Aluminum (0.5% Copper)
0.1 micron Titanium

Die Stepping: B

Pads per Die: 45

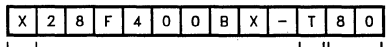
Intel Fabrication Process: ETOX™ III (minimum feature size: 0.8 microns)

Passivation: (from top surface)
2.3 microns B-Pyrox
1.1 microns Oxynitride

NOTICE: This document contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

4.0 DEVICE NOMENCLATURE



5.0 REFERENCE INFORMATION

Title	Order No.
28F200BX-T/B, 28F200BX-T/B 2-Mbit Boot Block Flash Memory Data Sheet	290448
28F400BX-T/B, 28F004BX-T/B 4-Mbit Boot Block Flash Memory Data Sheet	290451
AP-363, Extended Flash BIOS Design for Portable Computers	292098
ER-28, ETOX™ III Flash Memory Technology	204012
ER-29, The Intel 2/4-Mbit Boot Block Flash Memory Family	294013

6.0 REVISION HISTORY

Rev	Date	Description
001	3/94	Original version.
002	6/94	Replace Figure 1 to add GEL-PAK orientation information. General re-write of Section 2.0.

28F001BX-T/B 1M (128K x 8) CMOS FLASH MEMORY

SmartDie Product Specification

- **High Integration Blocked Architecture**
  - **One 8 KByte Boot Block with Lock Out**
  - **Two 4 KByte Parameter Blocks**
  - **One 112 KByte Main Block**
  - **T = Top Boot, B = Bottom Boot**
- **100,000 Erase/Program Cycles Per Block**
- **Simplified Program and Erase**
  - **Automated Algorithms via On-Chip Write State Machine (WSM)**
- **SRAM-Compatible Write Interface**
- **Deep-Powerdown Mode**
  - **0.05 μA I_{CC} Typical**
  - **0.8 μA I_{PP} Typical**
- **12.0V $\pm$5% V_{PP}**
- **High-Performance Read**
  - **70 ns Maximum Access Time**
  - **5.0V $\pm$5% V_{CC}**
- **Hardware Data Protection Feature**
  - **Erase/Write Lockout During Power Transitions**
- **ETOX™ III Nonvolatile Flash Technology**
  - **EPROM-Compatible Process Base**
  - **High-Volume Manufacturing Experience**
- **Intel SmartDie Product**
  - **Full AC/DC Testing at Die Level**
  - **0°C–80°C (Junction) Temperature Range**
  - **Available in 70 ns, 90 ns and 120 ns Access Times**

10

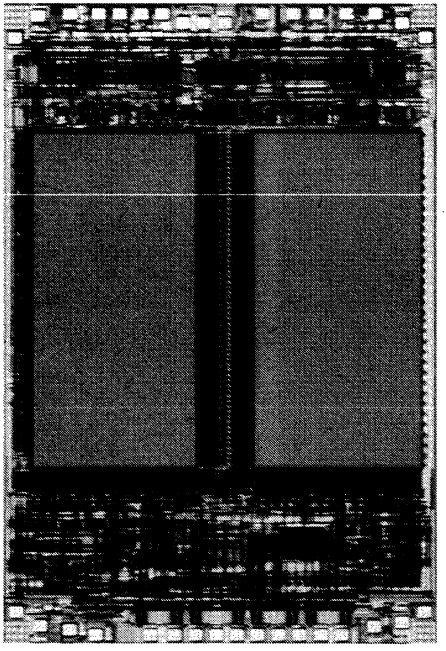
NOTICE: This document contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

REFERENCE INFORMATION: The information in this document is provided as a supplement to the Standard Package Data Sheet on a specific product. Please reference the Standard Package Data Sheet (Order No. 290406) for additional product information and specifications not found in this document.

---

*Other brands and names are the property of their respective owners.

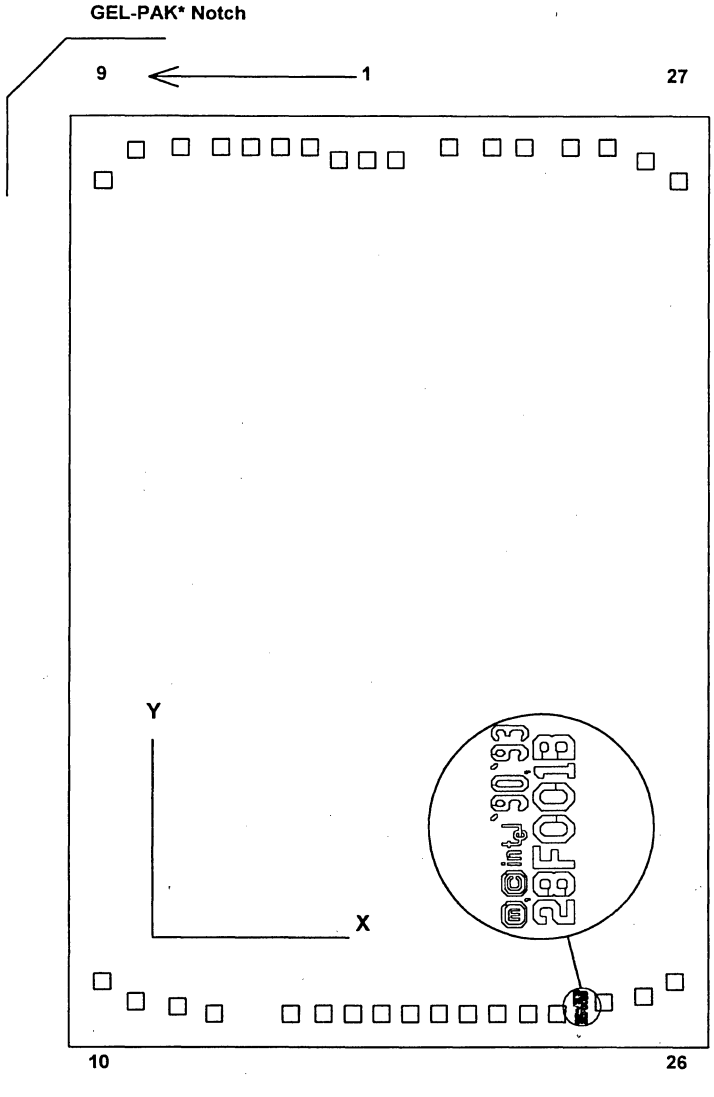
October 1994
Order Number: 271303-001



271303-1

28F001BX-T/B 1 M (128K x 8) CMOS Flash Memory Die Photo

1.0 DIE SPECIFICATIONS



NOTES:
Figure 1 indicates actual orientation of the die in the GEL-PAK* (shipping container). The 45° notch shown on the plot indicates the location of the 45° notch on the GEL-PAK.

Figure 1. 28F001BX-T/B 1M (128K x 8) CMOS FLASH MEMORY Die/Bond Pad Layout

1.1 Pad Description

Table 1. 28F001BX-T/B 1M (128K x 8) CMOS Flash Memory Bond Pad Center Data

Pad	Signal	Pad Center			
		(Mils = 0.001 in.)		(Microns)	
		X	Y	X	Y
001	V_{CC}	−5.7	106.1	−144	2695
002	V_{PP}	−13.0	106.1	−329	2695
003	A16	−20.3	109.3	−516	2776
004	A15	−27.8	109.3	−706	2776
005	A12	−35.4	109.3	−900	2776
006	A7	−42.9	109.3	−1089	2776
007	A6	−53.1	109.3	−1348	2776
008	A5	−64.4	108.7	−1636	2761
009	A4	−72.7	100.9	−1847	2564
010	A3	−72.5	−101.1	−1843	−2569
011	A2	−64.1	−106.1	−1629	−2695
012	A1	−53.5	−107.5	−1360	−2730
013	A0	−44.3	−109.3	−1125	−2775
014	DQ0	−24.9	−109.3	−631	−2776
015	DQ1	−16.5	−109.3	−420	−2776
016	DQ2	−9.0	−109.3	−228	−2776
017	V_{SS}	−1.7	−109.3	−42	−2776
018	V_{SS}	5.6	−109.3	143	−2776
019	DQ3	12.9	−109.3	328	−2776
020	DQ4	20.3	−109.3	514	−2776
021	DQ5	27.7	−109.3	704	−2776
022	DQ6	35.5	−109.3	901	−2776
023	DQ7	42.8	−109.3	1086	−2776
024	CE#	54.4	−106.3	1382	−2700
025	A10	64.5	−104.8	1639	−2662
026	OE#	72.5	−101.1	1841	−2569
027	A11	73.2	100.9	1858	2564
028	A9	64.9	105.9	1648	2690
029	A8	55.3	109.3	1405	2776
030	A13	46.0	109.3	1167	2776
031	A14	34.3	109.3	870	2776

Table 1. 28F001BX-T/B 1M (128K x 8) CMOS Flash Memory Bond Pad Center Data (Continued)

Pad	Signal	Pad Center			
		(Mils = 0.001 in.)		(Microns)	
		X	Y	X	Y
032	PWD#	26.1	109.3	664	2776
033	WE#	15.2	109.3	386	2776
034	V_{CC}	1.6	106.1	41	2695

Notes:
1. **X-Y pad coordinates represent bond pad centers and are relative to center of die.**
2. **The symbol "#" is used at the end of the signal name to denote an active low signal.**

2.0 INTEL DIE PRODUCTS PROCESSING

TEST PROCEDURE

Intel has instituted full-speed functional testing at the die level for all SmartDie products. This level of testing is ordinarily performed only after assembly into a package. Each die is tested to the same electrical limits as the equivalent packaged unit.

WAFER PROBE

Wafer probing is performed on every wafer produced in an Intel Fab. The process consists of specific electrical tests as well as device-specific functionality tests.

At the wafer level, built-in test structures are probed to verify that device electrical characteristics are in control and meet specifications. Measurements are made of transistor threshold voltages and current characteristics; poly and contact resistance; gate oxide and junction integrity; and specific parameters critical to the particular technology and device type. Wafer-to-wafer, across-the-wafer run-to-run variation and conformance to spec limits are checked.

The actual devices on each wafer are then probed for both functionality and performance to specifications. Additional reliability tests are also included in the probe steps.

WAFER SAW

Probed wafers are transferred to Intel's assembly sites to be sawed. The saw cuts totally through the wafer.

DIE INSPECTION

Upon completion of the wafer saw, the die are moved to pick and place equipment that removes reject die. The remaining die are submitted to the same visual inspection as standard packaged product. The compliant die are then transferred to GEL-PAKs for shipment.

PACKING PROCEDURE

Intel will ship all Intel die products in GEL-PAKs. GEL-PAKs eliminate the die edge damage usually associated with die cavity plates or chip trays.

The backside of each die adheres to the gel membrane in the GEL-PAK, eliminating the risk of damage to the active die surface. A simple vacuum release mechanism allows for pick and place removal at the customer's site.

Only die from the same wafer lot are packaged together in a GEL-PAK, and all die are placed in the GEL-PAKs with a consistent orientation. The GEL-PAKs are then sealed and labeled with the following information:

- Intel Die Products
- Intel Part Number
- Spec
- Customer Part Number (if applicable)
- Fab Lot Number
- Quantity
- Assembly Lot Traveler Number
- Seal Date
- ROM Code (if applicable)

NOTE:
GEL-PAKs require a Vacuum Release Station. For additional information about GEL-PAKs, contact Vichem Corporation.

INSPECTION STEPS

Multiple inspection steps are performed during the die fabrication and packing flow. These steps are performed according to the same specifications and criteria established for Intel's standard packaged product. Specific inspection steps include a wafer saw visual as well as a final die visual just before die are sealed in moisture barrier bags.

STORAGE REQUIREMENTS

Intel die products will be shipped in GEL-PAKs and sealed in a moisture-barrier anti-static bag with a desiccant. No special storage procedures are required while the bag is still unopened. Once opened, the GEL-PAK should be stored in a dry, inert atmosphere to prevent corrosion of the bond pads.

ESD

Components are ESD sensitive.

3.0 SPECIFICATIONS

Specifications within this document are specific to a particular die revision and are subject to change without notice. Verify with your local Intel Sales Office that you have the latest data before finalizing a design.

3.1 Flash Memory Handling Requirements

There are two key areas of concern for the X28F001BX. First, avoid exposing FLASH devices to ultraviolet light. Erasing the device under a UV light will erase device specific control information affecting device performance. Second, do not expose the device to temperatures above 350°C for greater than 10 minutes. Operation above this time/temperature envelope may cause damage to device reference cells.

3.2 Physical Specifications

Substrate Bias Condition: Float (Self-biasing to V_{SS}). Alternative is to drive V_{SS}.

Post-Saw Die Dimensions:

Mils: X = 161 ±0.5, Y = 234 ±0.5

See associated Die/Bond Pad Layout for X, Y orientation.

Die Backside Material (outer most layer first):
Polished Bare Silicon

Pad Passivation Opening Size:
Mils: 4.1 x 4.1 (single pads),
Microns: 105 x 105 (single pads),

Die Thickness: 17 ±1 mils

Minimum Pad Pitch:
Pads may not be evenly pitched. Minimum pitch is 185 microns (7.3 mils).

Bond Pad Metalization (outer most layer first)
0.9 Microns Aluminum (0.5% Copper)
0.1 Microns Titanium

Die Revision: A

Pads per Die: 34

Intel Fabrication Process: ETOX III (min. feature size 0.8 microns)

Passivation: (outer most layer first)
2.3 Microns B-Pyrox, 1.1 Microns Oxynitride

10

3.3 DC Specifications

ABSOLUTE MAXIMUM RATINGS*

GEL-PAK Storage Temperature ..... 0°C to +70°C
For other absolute maximum ratings, reference the 28F001BX-T/28F001BX-B data sheet, Order No. 290406.

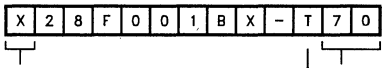
OPERATING CONDITIONS*

V_{CC}(Supply Voltage) ..... 5.0V ±0.25V
T_J(Junction Temperature Under Bias) ..... 0°C to 80°C

NOTICE: This document contains preliminary information on new products in production. It is valid for the devices indicated in the revision history. This specification is subject to change without notice.

***WARNING:** Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

4.0 DEVICE NOMENCLATURE



VALID COMBINATIONS:

X28F001BX-T70	X28F001BX-B70
X28F001BX-T90	X28F001BX-B90
X28F001BX-T120	X28F001BX-B120

271303-3

5.0 REFERENCE INFORMATION

Title	Order No.
28F001BX-T/28F001BX-B 1M (128K x 8) CMOS Flash Memory	290406
ER-20, ETOX™ II Flash Memory Technology	294005
RR-60, ETOX™ II Flash Memory Reliability Data Summary	293002
AP-316, Using Flash Memory for In-System Reprogrammable Nonvolatile Storage	292046
AP-341, Designing an Updatable BIOS Using Flash Memory	292077

6.0 REVISION HISTORY

Rev	Date	Description
−001	6/94	Initial release

intel®

11

Article Reprints

WP 3.6: Flash Solid-State Drive with 6MB/s Read/Write Channel and Data Compression

Steven Wells, Don Clay[1]

Intel Corporation, Folsom, CA/[1]Conner Peripherals Incorporated, Longmont, CO

Flash memory densities are sufficient for solid-state mass storage applications. Hard-drive emulation using flash memory provides a standardized system interface with characteristics superior to those of magnetic drives in read and seek performance, energy consumption, acoustics, ruggedness, and reliability. This 42MB 2.5" drive includes a thirty 8Mb device flash array, an embedded processor, and interface ASIC. The 0.7μm standard-cell ASIC contains drive interface circuitry, 4-port buffer manager, flash interface, and a Lempel/Ziv type hardware compressor. Flash device architecture is optimized for cost-effective high-density systems.

The dominant cost of the flash-based drive is in the flash device. Two issues have significant impact on flash device costs: 1) number of cells in the erase-block, and 2) required erase performance. Prior reported triple-poly and NAND device architectures closely match erase-block with drive- sector size and require high-performance erase before each sector write [1,2]. This flash drive supports a large erase-block size and reduces requirements on erase performance allowing use of lower-cost stacked-gate NOR flash devices (Table 1) [3]. A large erase-block reduces flash die size and complexity of decoders and drivers. Reduced requirements on erase performance reduce cost by simplifying the flash cell and enhancing yield.

The system block diagram is shown in Figure 1. The device array is organized into pairs for increased performance. The embedded processor handles drive commands and manages the flash. A buffer RAM provides drive-interface compatibility and allows caching.

A linear write algorithm executed by the embedded processor writes each sector to the next available location in the flash array and notes the sector number. A prior copy of a sector stored in the flash array is marked "old" and is no longer referenced. Reclamation of old sectors is managed by the embedded processor transparent to the drive interface by identifying erase-blocks with old sectors, moving remaining sectors to other blocks and initiating erase. Measured program/erase cycling with this algorithm is shown in Figure 2. Optimum erase-block size for stacked gate NOR flash devices minimizes the time required to reclaim old sectors (Figure 3).

A custom interface ASIC is designed for maximum transfer performance from flash to drive interface. A 20MB/s circuit transfers data from flash to the buffer RAM. A 4-port prioritized buffer controller interfaces the flash transfer circuits, the drive interface, and the embedded processor with sustained 6MB/s on both drive interface and flash media. The interface controller is state-machine controlled to allow the embedded processor to focus on acquiring the next sector or flash reclamation. The drive is capable of sustained 4MB/s random read transfers.

Zero spindle-energy provides 30x energy savings over conventional magnetic drives. This drive quickly enters and exits lowest power oscillator-disabled sleep modes. An intermediate energy-saving mode disables ASIC clock distribution during momentary idle times; a data request re-enables it within 50ns. Figure 4 diagrams the oscillator control and clock distribution circuit. $\overline{\text{REQUEST}}$ strobes the oscillator or clock disable request into latches L1 and L3. Latch L1, which stores the clock on/off request (CLK_EN), disables the ASIC clock via embedded processor command ACTIVATE A host interrupt re-enables it through latch L2. Latch L3 stores the embedded processor oscillator-disable request from OSC_EN. Latches L4-L8 ensure glitch-free clock-signal propagation. The ASIC vendor optimized latch elements reduce metastability.

A Lempel/Ziv data compressor is integrated into the read/write channels to enhance read/write performance, reduce energy consumption, increase density, and enhance effective buffer bandwidth. The compressor operates on 512B sectors and achieves an average compression ratio of 1.6x at a sustained 6MB/s for read and write. The linear write algorithm combined with large erase blocks is key for efficiently storing variable-size sectors. Increased density is achieved through drive interface protocol enhancements. Sectors that generate compression ratios <1 are stored uncompressed.

Overall system performance is compared to a typical 2.5" drive in Table 2. Figure 5 shows drive read timing with parallel host, sector buffer, and flash transfer. A photograph of the system is shown in Figure 6.

Acknowledgments

This work was made possible by S. Anderson, R. Garner, T. Kendall, R. Hasbun, S. Domonkos, S. Barbarich, M. Powell, and D. Edwards. The authors thank J. Squires, P. Jacobs, D. Pashley, B. McCormick, M. Winston, and J. Weisenstein for vision and support.

References

[1] Mehrotra, S., et al., "Serial 9 Mb Flash EEPROM For Solid State Disk Applications", Symposium on VLSI Circuits, pp. 24-25, June 1992.

[2] Itoh, Y., et al., "An Experimental 4Mb CMOS EEPROM with a NAND Structured Cell", ISSCC DIGEST OF TECHNICAL PAPERS, pp. 134-135, Feb. 1989.

[3] Kynett, V., et al., "A 90ns 100k Erase/Program Cycle Megabit Flash Memory", ISSCC DIGEST OF TECHNICAL PAPERS, pp. 140-141, Feb. 1989.

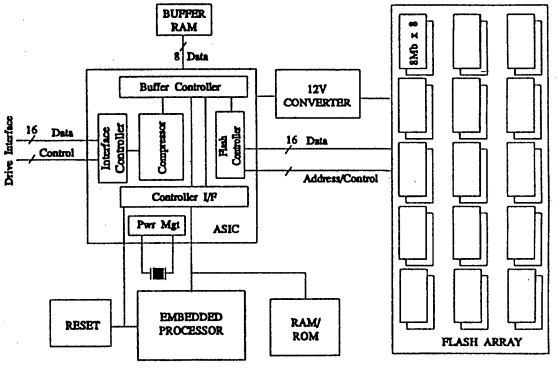


Figure 1: System block diagram.

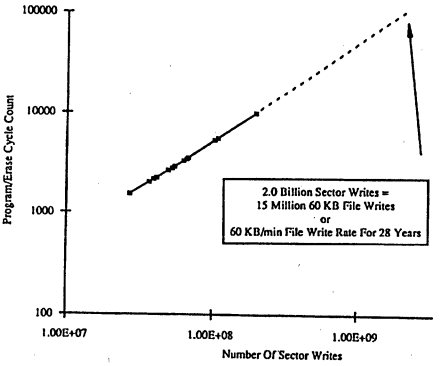


Figure 2: Device cycling versus sector writes.

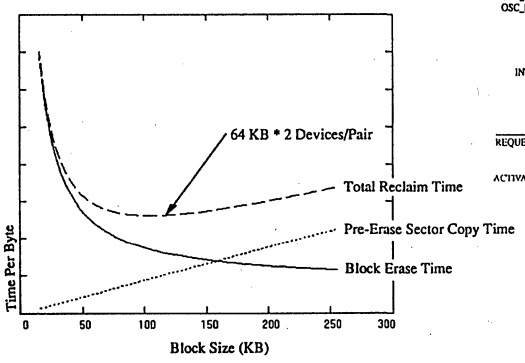


Figure 3: Optimum block size to minimize reclaim time.

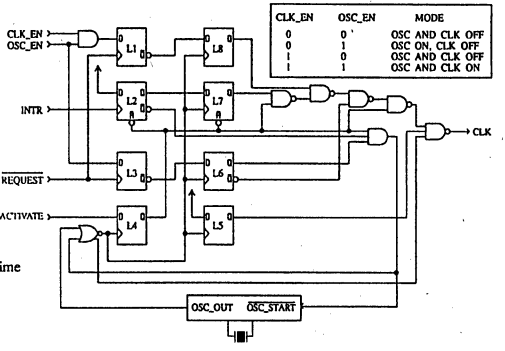


Figure 4: Oscillator control and clock distribution.

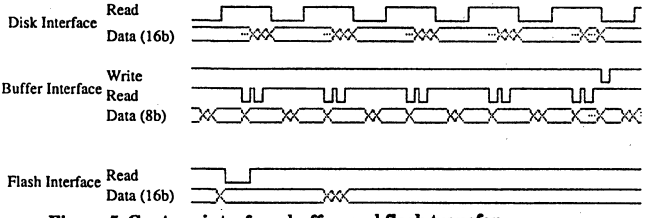


Figure 5: Capture interface, buffer, and flash transfer.

Figure 6: See page 262.

Feature size	0.8µm
Cell	7.25µm² stacked-gate NOR
Organization	8Mbx8
Erase block	64kB
Die size	102mm²
Read throughput	70ns/B
Program throughput	111kB/s
Erase throughput	64kB/s

Table 1: Flash device features.

	This System (read/write)	Mag. Drive
Interface rate	6MB/s, 6MB/s	8MB/s
Compression	Yes	No
Media rate	9.6MB/s, 0.36MB/s (with compression)	1.6MB/s
Spinup time	3ms, 6ms	7 - 20s
Shock, op/non-op	1kG/1kG	10/100G
Seek, avg/trk-trk	<0.25ms, 0.25ms	15ms,2.5ms
Rotational latency	0ms	8ms
Acoustics	Silent	34dBA

Table 2: Drive performance.

WP 3.5: A 30ns 256Mb DRAM with Multi-Divided Array Structure
(Continued from page 51)

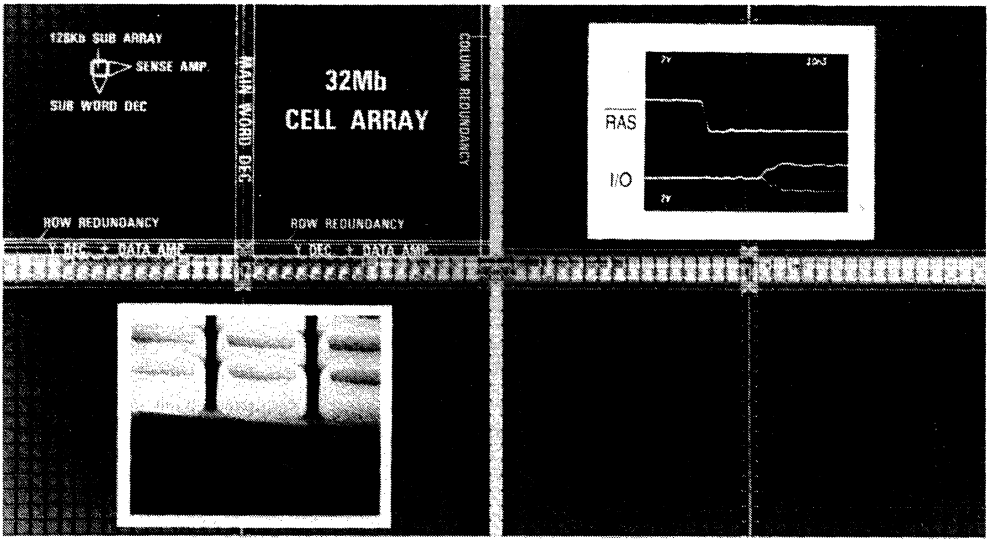


Figure 7: 256Mb DRAM chip micrograph, memory cell cross-section, and measured waveforms.

WP 3.6: Flash Solid-State Drive with 64Mb/s Read/Write Channel and Data Compression
(Continued from page 53)

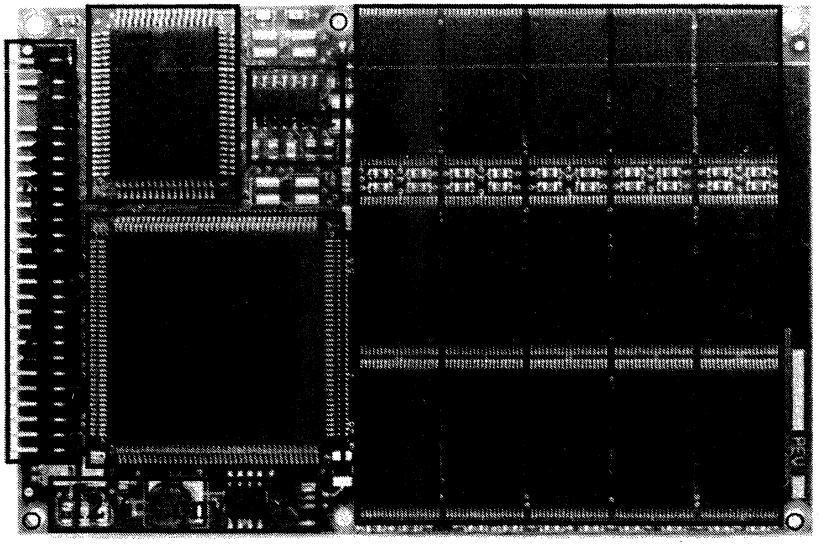


Figure 6: Solid-state drive micrograph.

intel®

ARTICLE REPRINT

AR-711

FLASH: BIG NEWS IN STORAGE?

ARE DISKLESS COMPUTERS THE WAVE OF THE FUTURE?

by Walter Chen

Imagine, in the future, when you turn on your personal computer and the next sound you hear is...silence. There is no familiar whir of the hard drive greeting you, just the low murmur of the power supply. Inside, the rigid disk has been replaced by the solid-state advantages of flash chip technology.

Sure, it's a conceivable scenario, but don't expect it to be in the near future. Although flash chip technology has already found its way onto some desktops and portable computers—used for either the BIOS or as a hard drive replacement—even flash manufacturers don't believe their product will eventually make hard disks obsolete. And with competing technologies such as SRAM, EPROM and EEPROM already established, flash will have to shine above this group before it can take on the big boys.

"We have all kinds of way of niching against disk technology in the desktop market, but we're not pursuing it for the time being because we don't see that as high growth," said Bruce McCormick, the marketing director for Intel's flash card. Instead, McCormick added, Intel is focusing more on low-end computers and consumer products as the future bread and butter of flash.

Flash technology, which was pioneered by Toshiba in the mid-1980s, uses non-volatile solid state memory chips. Because it retains data when power is turned off and because it's rewritable and rugged, flash chips have become especially appealing in some applications. In portable environments, for example, where small form factor and durability are crucial requirements, flash appears to be the next revolution.

But flash also has considerable disadvantages, the most noted of which is price. Still a technology in its infancy, flash chips cost about $30 per megabyte, compared to hard drive

Reprinted with permission from Storage Magazine, November/December 1992 edition Order Number 295078-001

prices that have fallen to as low as $2 to $3 per megabyte. Additionally, the capacity of flash chip drives, or cards, remains expensive compared to conventional storage devices. Twenty megabytes is the upper capacity of flash cards now, while even some small form factor hard drives have capacities of a gigabyte or more. And even with expected price drops in flash technology as well as improvements in capacity, some industry observers say they don't believe flash will pose a serious threat to drive makers in the near future.

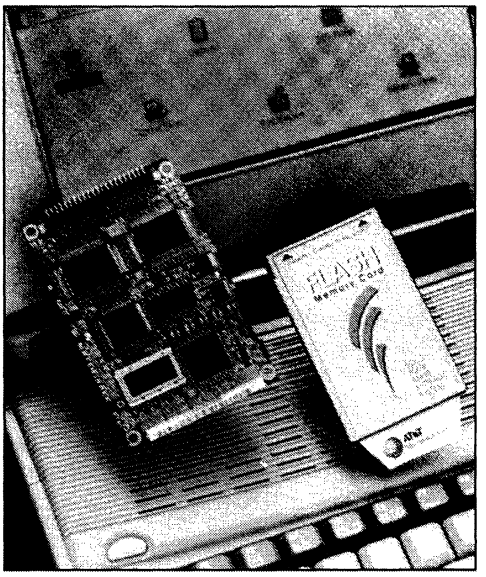
"I do not see flash as a hard drive killer. I believe it will put mass storage capacity in very small things," said Walt Lahti, the vice president of In-Stat, a market research firm based in Scottsdale, Ariz.

Lahti said there is no doubt that flash will be able to find a market despite the vast variety of drives and other solid state chip makers. However, he added, drive makers will always have the advantage of larger capacity.

In addition, flash has a limited life, the majority of flash products on the market today requires two power supplies and unlike some other solid state technologies, data can't be written to the chip on a byte-basis. Until recently, changing data on a flash chip required erasing and rewriting the entire chip. AT&T recently introduced a flash card that allowed data to be erased in block sizes of 256 bytes.

Even head-to-head on form factors, hard disks have become even smaller, with the introduction of 1.8- and 1.3-inch drives. Cards complying with the Personal Computer Memory Card International Association (PCMCIA), flash cards measure 3.3 by 2.1 by .1 inches.

But what flash has to offer is also unmatched. Similar to EPROM (erasable ROM) and EEPROM (electrically erasable ROM), flash cells are smaller and use just one transistor per memory bit. In a flash chip, data storage is achieved by trapping electrons in a capacitor. They stay there even after electricity has been turned off. To reprogram the chip, voltage is reversed and the capacitor is purged through what's called a tunneling process. According to Intel, data is retained for as long as 100 years.



AT&T's flash card

In other technologies, like ROM, information is stored in a fixed bit pattern that can't be altered. EPROM is similar to ROM except that data can be changed by exposing the chip to ultraviolet light. In EEPROM, strong electrical currents, rather than ultraviolet light, are used to change the data. EEPROM is also more expensive and slower than flash.

While RAM—both static and dynamic—can also act as storage devices, they are volatile, meaning data are lost the moment electricity is turned off. While batteries are sometimes used to keep data contained in SRAM, this technology still doesn't offer the advantage of flash. DRAM, however, is not threatened by flash because it is needed for data manipulation, which flash is unable to do.

Similarly, flash is faster than small form factor hard drives, weighs less and consumes less power. It's also more rugged than hard drives, whose mechanical parts are vulnerable to shock.

Although the cost of flash is roughly 10 times that of rigid disks now, those prices are falling at a dramatic rate. For instance, Intel said its cost-per-megabyte for flash was $640 in 1988. It also predicts that by the end of the century the per-megabyte cost will be below $1.

Additionally, while chip density is currently at 8Mbit, a 16Mbit chip is soon expected, allowing 40MB cards to be made, said Glen Riley, the marketing applications manager for AT&T's flash memory division. The telecommunications giant jointly worked on developing a flash card with SunDisk and has introduced a 20MB card that can be paired up in an IDE

controller to provide up to 40MB of storage capacity.

He said that while current chips only have densities of 8Mbits, he predicts 16Mbit chips available by 1994, 32Mbits by 1997 and a 64 or 128Mbit chip by the end of the century. Similarly, he said he sees the cost drop by half every two or three years.

If, indeed, flash costs can fall dramatically, the technology is likely to show up in more than just computers. It's likely to be found in everyday consumer goods, printers and other electronics requiring mass storage in a small form factor.

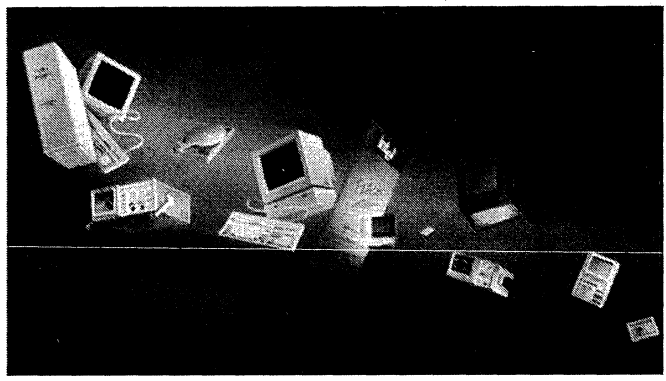
Already, flash cards are appearing in a number of products, including several notebook computers, desktop computers for storing BIOS and medical equipment.

For example, Hewlett-Packard recently introduced a flash card system to its 95LX palmtop computer, which weighs 11 ounces. The card, produced by SunDisk and which comes in a 20MB size, gives the computer far more storage than its SRAM memory, which had a maximum capacity of 1MB.

Richard Kirby, the developers program manager with HP's portable computing Corvallis division, said flash provides much more storage than SRAM and allows users to have access to more applications.

"I think there are things that can't be done with the small amount of memory that SRAM provides," he said, "and flash will increase the amount of applications."

Ironically, HP, whose recently introduced Kittyhawk disk drive will compete with flash cards, is just one of the handful of companies using flash in their products. Kittyhawk, which is the industry's first 1.3-inch hard drive, is the smallest disk drive on the market. Psion last year introduced a flash-based laptop that, while disappointingly



underpowered, nevertheless was one of the first to market with a flash-equipped computer.

Already, some 20 companies are looking to flash as the next cash cow. Companies like AT&T, SunDisk, Advance Micro Devices, Fujitsu and others are trying to catch up to Intel. For instance, AMD and Fujitsu agreed in July to develop both EPROM and flash devices.

INTEL THE LEADER

Still, they have much ground to make up to match Intel, the clear leader. The company manufactures several products, including memory cards with capacities of four, 10 and 20 megabytes. Lahti said the chip maker last year dominated 75 percent of the

MICROSOFT FILE SYSTEM ALLOWS FLASH TREATED AS DISK DRIVES

Microsoft, whose MS-DOS has become the standard operating system for the PC industry, appears poised to lead the development of flash applications as well with its flash file system.

The file system contains essentially three drivers that make flash cards appear to users as if they were magnetic disks. Through hard disk emulation, Windows and DOS applications can be executed on flash cards without extensive modifications of the software, said John Kechejian, the product manager for MS-DOS at Microsoft. This also aids developers of software, he added.

The drivers are loaded into the CONFIG.SYS file of DOS. One driver interfaces with the operating system and acts as a translator for a second driver, which carries out the command on the flash cards. A third driver, said Kechejian, enables compression on the flash card.

Kechejian said the flash file system is designed to complement flash cards' unique characteristics.

Because the life of flash chips is limited to about 100,000 write-erase cycles, the drivers have to make sure data are placed evenly throughout the card to ensure an even wear of the chips; otherwise, "hot spots" are created that lead to premature failures.

$130 million market, a market that is expected to see explosive growths in coming years.

Market Intelligence, a market research group, said flash sales this year are projected to account for six percent of the total market for small form factor storage devices, which include micro disk drives and flash cards. But by 1998, flash is expected to dominate, growing to 41.1 percent of the market, which is expected to grow to $11.6 billion by then.

"It's an evolutionary thing. We had ROM and then EPROM and then EEPROM," said Sandeep Maheshwari, an analyst with the Mountain View, Calif. firm. "Flash is certainly a superior technology. It takes the benefits of both EPROM and EEPROM."

Maheshwari said he sees flash eventually muscling out low-capacity hard drives and will compete with drives in the 150 to 200MB range. With such memory hogs like UNIX, Windows and other graphic user interfaces becoming more popular, he said there will be greater demands for mass storage.

FLASH'S MARKET

Flash won't, however, replace all hard drives, say analysts and even some flash makers.

"Definitely Fujitsu believes there's a future in flash. My only question is whether it's going to replace disk. We don't think it's going to happen," said Alex Goldberger, the director of strategic marketing at Fujitsu. "Personally I keep seeing that for every quantum leap in solid state storage, there's a quantum leap in rigid disk storage."

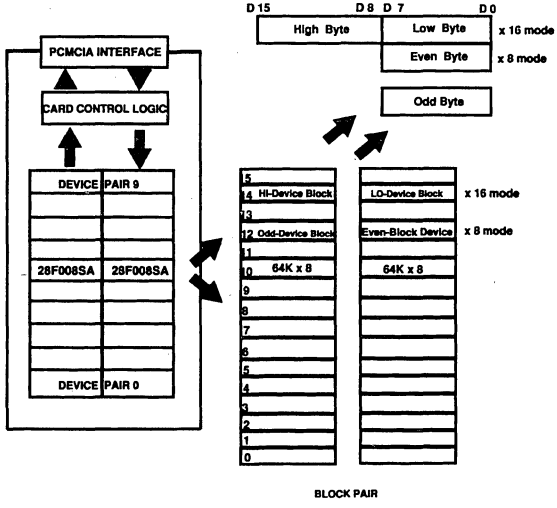
And as flash prices fall, observer say, prices for hard drives will do the same. And some believe that flash will simply never be able to catch up to hard disk drive on a per-megabyte cost. If anything, flash is poised to displace EPROM, some say.

"The EPROM market will be the one most heavily impacted," Lahti added, pointing out that once flash supplants EPROM, its next battle will likely to be with EEPROM chips. "The EEPROM versus the flash (card) is the most direct socket-for-socket battle that will take place."

But the optimistic outlook on flash is also based on anticipated high growth in the portable computing market, which Riley said has grown to expectations, adding that "1992 was a disappointment for us in the mobile computing year because it didn't take off as fast as we thought." He said that there are probably less than 50,000 portable computers using flash now. So if the proliferation of portable computers continues to lag, flash may be even less of a threat to hard drive.

That's why, McCormick added, Intel is interested in placing flash chips in consumer electronics, products that enjoy a much broader customer base. He sees items such as cameras, medical instrumentation, flight recorders and others as virgin territory for flash, holding much promise.

"There will always be cheaper archival technology (than flash) for some time," he said. "All we're saying is on the low end flash can be your temporary storage. On the real low end like a camera...it can end up being an archival device as well. S



Intel's Series 2 Flash Card Architecture Overview

FLASH MEMORY: MEETING THE NEEDS OF MOBILE COMPUTING

1992 Pen-Based Expo
Brian Dipert
Intel Corporation

INTRODUCTION

Have you noticed the diversity of approaches that computer manufacturers are taking as they develop and introduce their initial pen-based computers? Different CPUs, different combinations of input devices, different types and sizes of mass storage subsystems, different product "packaging", size and weight.....the industry is struggling to first define just what is a pen-based computer for their specific target markets, and then provide product features and capabilities to match. This paper will present flash memory in all its forms as a key technology that will help bridge the gap between the pen-based computers of today and their "ideal" counterparts of tomorrow.

Industry analysts are predicting an impending market demand explosion for portable computing platforms with pen-based human interfaces. This customer interest will partially come from existing users, who value the capabilities of fully mobile computing. Equally enticing to hardware OEMs for their total market growth potential, however, are the new users, who today are not computer "literate", and who will purchase these machines precisely because they do not operate like today's computers. These potential customers are intimidated by today's keyboards, bulky and heavy hardware "packaging" and cryptic command interface.

Tomorrow's computing consumers, who will continue the market explosion begun in 1975 with the introduction of the Altair, don't understand computers in great detail and don't want to have to learn. They don't want a tutorial in file system fundamentals. They want unlimited variety in the types of data they can interchange with each other, and straightforward, intuitive methods to accomplish this interchange. Finally, they won't accept (within reason) limitations in the environments and ways that they operate their computing platforms. Look at the telephone for a model of the ideal "computer" of tomorrow; it should be just as straightforward to operate. Why don't the vast majority of VCR owners use the recording capability of their units, and why do the clocks on these VCRs "blink" annoyingly with the familiar "12:00" display? Because the mechanism to program a VCR, or to reset its clock after a power outage, are not straightforward or intuitive!

The computer industry's challenge to satisfy the needs of tomorrow's users is indeed a tall one. It requires fundamental reshaping of the way the software industry architects its user interfaces, to make a computer less a "puzzle" and more a "tool". The goal of the computer hardware industry is equally ambitious. This objective demands a fundamental redesign of the computer architecture, away from the "desktop" model which has served the industry for the last 10 years (and the "mainframe" model before that), and toward a new model for computing in the '90s and beyond. The central improvements needed center around the 4 "P"s of portable computing: Performance, Portability, PC Compatibility and Pen Interface.

Performance

Tomorrow's computing subsystems will handle increasingly complex data types (including still and full motion video images, and hi-fidelity stereo sound), and manipulate this data faster and more "invisibly" than ever before. This demands improvements not only in the CPU itself, but also in the memory subsystems and how they are accessed. For today's PC users that will upgrade in the future, the goal is replicate the desktop PC on the road....or totally replace it! Woven into this hunger for *greater performance* is the diametrically opposed need for greater and greater battery life, and smaller and smaller batteries. This leads to the next "P".....

Portability

Today's mobile computers have come a long way from 1985's Compaq Portable. However, when the author travels on the road, his notebook PC with extra battery pack and AC power supply still weighs more than 15 pounds, is over 2" thick and requires its own separate carrying bag. A one-way plane flight across the United States is more than enough time to drain both batteries. Today's portable PCs are in most cases portable only in that they can be **transported** from one location to another; they must still be **operated** on a stable, flat, non-mobile surface like a table top. The goal of pen-based computing is to provide a comparable usage model to today's pencil and pad of paper. To do so requires "pen-and-paper-like" attributes: much *longer battery life*, and much *lighter* and *thinner* packaging.

PC Compatibility

Today's installed volume of Intel-based personal computers numbers over 100 million units. Compatibility and data interchange capability with this installed base is essential. This does **not** necessarily mean that tomorrow's mobile computers must run the same MS-DOS*, Windows* or OS/2* as their desktop brethren. What it **does** mean, however, is that file *read/write adaptability and exchange* must be preserved. Additionally, a familiar software interface will assist users that work with both types of computers. One way to implement this latter goal is thru a.....

© INTEL CORPORATION, 1992

Order Number: 295082-001

Pen Interface

A recent article in PC Week, reviewing today's pen-centric hardware, software operating systems and extensions, reported that, "If the point of using these systems is to input information by writing it, the technology (today) won't work"[1]. Quotes in the same article from test group users reflect this opinion. "I was just amazed at the difficulty in using the pen-based system....It was unresponsive and time-consuming, and I spent more time than I should have inputting answers to questions"[2]. "I spent 35 minutes to an hour just trying to record my name.....I don't know who would use it—you'd go crazy. I can sit down and key a lot faster than I write"[3]. "The purpose of writing is to write at your normal pace....If I have to slowly write A.....B.....C....., I may as well type"[4]. *Handwriting recognition software* of the future must continue to improve in its handstroke translation power and ease of use.

HOW DO WE ACHIEVE THE IDEAL?

The last several paragraphs have spoken in generalities about the attributes of the ideal mobile computer. Now for some hard facts; how do we get there from today? What are the "weak links" in today's mobile computer architecture that must be improved or replaced to achieve the ideal?

Make it Lighter

The heaviest subsystems in a computer are the keyboard, the battery pack and power supply, and the floppy and hard disk drives. Resolving the first item is easy: remove the keyboard (or make it an option) and replace it with a lightweight pen interface. Efficient power management, both in hardware and software, will minimize system power draw and enable lighter and smaller batteries and power supplies. Finally eliminate the floppy disk drive and remove (or reduce the density of) the hard disk drive, achieving both aims via alternative non-rotating memory media. More on this later....

Make it Thinner

The two biggest contributors to the thickness of today's portable PCs are the floppy and hard disk drives. Again following the recommendation above, removing the FDD and HDD is the means of achieving this ideal. An array of 1.2 mm tall high-density memory components spread out on the system motherboard will make the <1" thick mobile computer a reality.

Give it Longer Battery Life

The screen is one of the largest power consumers in a computer. Recent (and future) advances in screen technology promise tremendous advances in this area. Continued CPU and chipset innovations, coupled with software "hooks", will produce more intelligent power management as semiconductor manufacturers drive innovation and on-chip integration in this area. Today's system DRAM requires constant refresh to maintain valid data contents (even during SUSPEND modes). Reduced reliance on DRAM as a main system memory will result in system power savings. Finally, the motors in today's rotating mass storage media are inherently power-hungry. Solid state mass storage is the key to eliminating this source of shortened battery life.

Give it Higher Performance

Ever-increasing silicon "intelligence" will result in future CPUs with performance measured not in MIPS, but in MIPS/watt. Beyond the raw speed improvements of the processor "engine", a transition to a main memory subsystem consisting of a direct-execute, nonvolatile, updateable code array eliminates the slow seek and rotation delay of code loaded from the HDD.

Make it More Rugged

Continued advances in composition and manufacturing technology will improve display screen ruggedness, an "Achilles Heel" of today's portable PCs. Rotating mass storage media is relatively fragile. In comparison, solid state devices are inherently rugged; ready to take the punishment of the truly mobile computing environment. A comparison of flash memory card specifications (1,000 G operating shock resistance) to HDD specs (10 G) shows a 100x improvement. Will tomorrow's mobile computer user wait for the HDD to park its heads before tossing his/her "tablet" PC on the front seat of the delivery truck? How valuable is the data stored on this "tablet"?

Make it Easily Upgradeable, Enhanceable

Consumers value upgradeability, to the latest version of their favorite software, to the latest power management BIOS, to the latest set of "whizbang" features and capabilities. A very good way to ensure long-term customer loyalty is to design a system that is not obsolete 6 months after he/she buys it. Customers also don't appreciate buying initial production products full of "bugs" that are difficult, expensive or impossible to fix in tomorrow's compact, integrated systems. Easy system upgradeability and enhanceability, done by the user instead of an expensive technician, is a powerful customer service differentiator in a market of many competing alternatives.

Give it an Integrated Pen Interface

This last point is an automatic given. What better way to increase the number of computer users than by providing them an intuitive interface that they've used since childhood; the pen? However, continued innovation both in pen hardware and software are essential to make this goal a reality.

FLASH MEMORY: ENABLING SUPERIOR MOBILE COMPUTING

The past several pages have described the ideal mobile computer, and the improvements to today's hardware and software designs that will lead to this ideal. What has any of this to do with flash memory? Flash memory is a fundamentally new memory approach that is a key enabling technology for superior mobile computing products. In a few words, flash memory:

- ❒ Makes mobile computers lighter and smaller
- ❒ Makes mobile computers faster
- ❒ Enables PC software compatibility, and
- ❒ Enables easy upgradeability and enhancement.

The next few pages will detail flash memory usage in specific application areas. First, however, it is useful to explain flash memory and its characteristics in some depth.

What is Flash Memory?

Flash memory, first introduced by Intel Corporation in 1988, is at its core derived from fundamental EPROM technology. It combines ideal attributes of several of today's mainstream memory approaches. Flash memory (as shown in Figure 1) combines the:

- ❒ High speed of DRAM
- ❒ Nonvolatility of hard disk drives and floppy disk drives
- ❒ Updateability of RAM or EEPROM, and
- ❒ High density of ROM

Comparative cell diagrams for an EPROM cell and ETOX™ flash memory cell are shown in figure 2. Clearly, ETOX flash memory and EPROM share similar cell structures; this makes flash memory highly manufacturable and scaleable. Their bit program mechanisms are identical. However, the narrower gate-to-substrate oxide thickness of a flash memory cell enables electrical erasure, versus the UV erasure of EPROM.

Flash memory is programmable (writing data "1"s to "0"s) on a bit-by-bit resolution. Erasure (changing "0"s back to "1"s) is accomplished on a block-by-block level (as small as 4Kbytes on some devices). A Command Register architecture results in SRAM-like command write timings to flash components, and newer devices integrated a Write State Machine to automate and internalize program and erase algorithms.

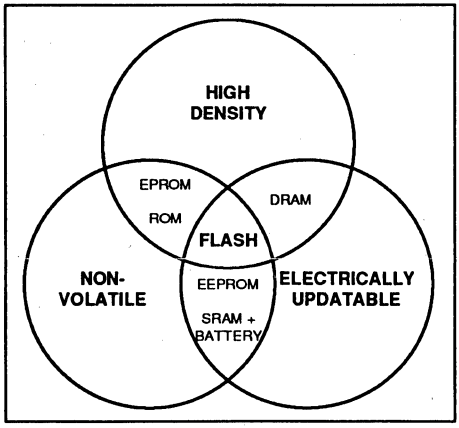


Figure 1. Flash Memory, the Optimum Nonvolatile Memory

Characteristics of today's state-of-the-art flash memory components and cards are shown in Table 1.

WHERE CAN FLASH MEMORY BE USED IN MOBILE COMPUTING?

Flash memory usage centers around the following four applications:

- ❒ BIOS storage
- ❒ The Resident Flash Array
- ❒ Flash Memory Cards, and
- ❒ The Solid State Drive

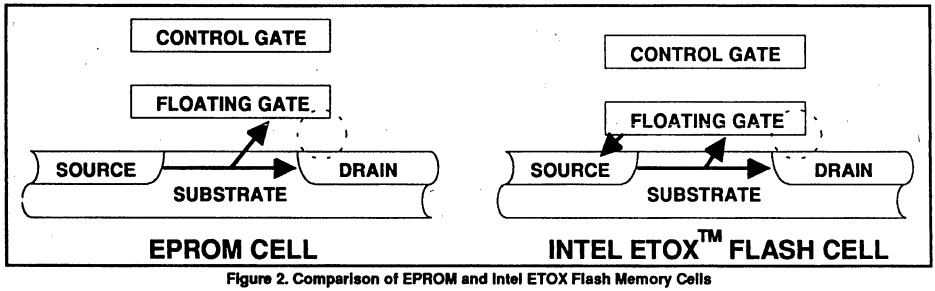


Figure 2. Comparison of EPROM and Intel ETOX Flash Memory Cells

Table 1. Flash Memory Characteristics (Current as of Paper Publication Date)

Read Access Time	60ns Maximum (Component)
	200ns Maximum (Cards)
Bit/Byte/Word Write Time	9 µsec Typical (SRAM-Like Command Write Speed)
Block Erase Time	1.6 sec Typical (64KByte Block)
Minimum Block Erase Cycles	100,000 Cycles
Density	32KBytes to 1MByte (Components)
	1 MByte to 20 MBytes (Cards)
Read (Operating) Voltage	5V ± 10%, 3.3V ± 0.3V
Program/Erase Voltage	12V ± 5%

BIOS

The term BIOS stands for Basic Input-Output System code. As Figure 3 indicates, BIOS is the lowest level software interface between the system software and hardware. BIOS is in a sense the "glue" that binds multiple diverse hardware implementations to a common set of software operating systems. In the IBM-compatible world, the BIOS is viewed as the basis of PC compatibility. The BIOS specifically controls such hardware subsystems as the CPU and coprocessor, its chipset, the graphics and main memory subsystems, keyboard/mouse/pen, networking interface and secondary storage subsystems. Also often lumped under the general "umbrella" of the BIOS is software such as power management code, "docking station" software and any resident "ROM"-executable operating systems and applications. It should be clear just how crucial proper BIOS code execution is to correct operation of the computer!

What is the current and future environment under which a BIOS is developed and operated? Today's hardware designs are becoming more and more integrated, with increasingly higher functionality CPUs and chipsets. In some aspects, this tends to make hardware design a "simpler" task.

In conjunction with this hardware trend, computer users are demanding more powerful and flexible computers. This results in more and more complex software of all kinds, including the system BIOS. Simpler hardware design, coupled with more complex software requirements, means that the BIOS development often gates time-to-market for new computer designs. More intricate software equates to greater potential for software "bugs", especially in the last-minute rush to bring a new system to market ahead of the competition. As hardware becomes increasingly integrated, the resultant systems are more and more compact, and difficult and expensive, if not impossible, to disassemble if the BIOS component requires replacement.

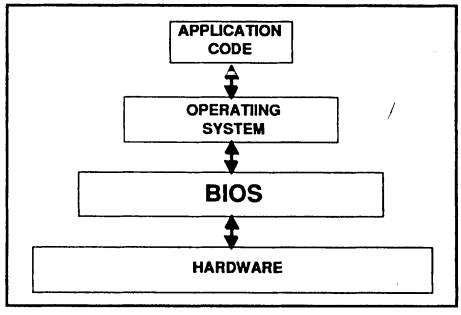


Figure 3. BIOS; Lowest Level Hardware/Software Interface

Common BIOS storage solutions include ROM (Read-Only Memory) and EPROM (Eraseable Programmable Read-Only Memory). ROM has a low per-device cost to system manufacturers, and the BIOS code is "burned" in the device at the semiconductor vendor fab. However, a minimum ROM order quantity of multiple-thousand devices is often required. ROM is not reprogrammable, so if a "bug" is discovered in the "burned-in" code, the devices must be scrapped. ROM is also not reprogrammable once installed in a system. If a "bug" is discovered once the computer is in a customer's hands, the unit must be disassembled to replace the defective ROM code. In-system BIOS enhancements are also not possible. EPROM, while more expensive than ROM, is shipped "blank" from the semiconductor vendor and programmable by the computer manufacturer. Again, however, it is not in-system reprogrammable and must be removed and replaced if the code contained inside is found to be faulty. *What alternative exists that allows factory programming (and reprogramming) and easy in-system field update?* Flash memory!

Flash memory, by virtue of its electrical programming and erasure, is easily updated under system software control while physically connected to the computer motherboard. This allows, for example, a manufacturer to download test code as the system moves down the manufacturing line, and reprogram the exact BIOS required as the computer leaves the factory. One hardware design can therefore easily be customized to match the needs of multiple markets. Finally, by simply running an "update" routine and accessing code from an OEM-supplied diskette or electronic BBS, a customer can easily update his/her system BIOS to remove initial production "bugs" or enhance capabilities. The result is a longer system lifetime, postponed obsolescence and long-term customer satisfaction.

Intel's BootBlock product line has been architected specifically with features that satisfy the requirements of BIOS storage in mobile computers. For more information on these devices, please reference the Additional Literature section at the conclusion of this paper.

Resident Flash Array

Today's memory subsystem is shown in Figure 4a. The process by which a CPU accesses a byte of code is today a lengthy and complex ordeal. The code is first accessed from the hard disk drive (after waiting a relatively long time for the drive rotation and seek delay). It is copied from the HDD to system DRAM, and from there to the high-speed SRAM cache. Then, and only then, can the CPU fetch and execute it.

There appears to be a redundancy between the "memory" systems of the HDD and the DRAM array. Why are both needed? Simply, they exist to counterbalance each other's shortcomings. DRAM's read access speed is relatively fast, but it is a volatile memory (in other words it loses its data if it loses power or is not refreshed). SRAM is similarly volatile. The HDD, while fully nonvolatile, has a relatively very slow read access time.

Figure 4b suggests an alternate approach, the RFA (Resident Flash Array). Flash is an ideal memory technology for mass storage in that it combines the fast read access time of DRAM with the nonvolatility of a hard disk drive. A high-density array of flash memory, storing "ROM"-executable code such as Microsoft*'s MS-DOS* 5.0 ROM Version (currently in production) and Windows 3.1 ROM Version (currently in betasite testing), provides high-performance "instant-on" execution and task switching, and much lower system power consumption by replacing the rotating HDD and a majority of constantly-refreshing DRAM. The annoying delay of staring at the Windows "hourglass" while code slowly loads from the HDD to DRAM is eliminated!

Why use flash memory versus ROM? Flash memory's in-system updateability is as invaluable here as it is when used for BIOS storage. When the next version of a user's favorite "ROM"-executable application is released, flash memory provides him/her the opportunity to easily upgrade the RFA. It is important to note that flash memory (by virtue of its block-erase characteristics) can displace, but will not totally replace, system DRAM. DRAM will continue to be utilized where full bit alterability is required (such as in manipulation and storage of application temporary data, video graphics data or interrupt vector tables). However, for application code storage where read-only capability is sufficient, the RFA provides a superior solution to the redundancy of DRAM and disk.

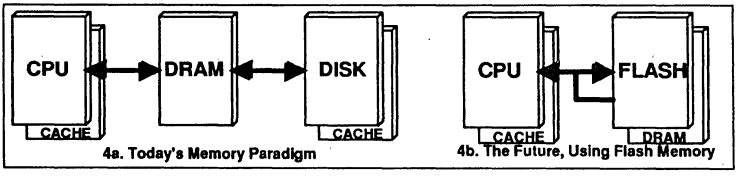


Figure 4. Flash Memory Revolutionizes the Architecture of Computing

Where "ROM"-executable code is not available, the RFA concept is equally useful in a resident solid-state "drive" configuration. In this respect the RFA "HDD" is not unlike today's "RAM drives" made up of DRAM and configured via software. Of course there is one important difference; a "Flash Drive" is completely nonvolatile and won't lose its data when power is removed! By putting the necessary code in the system BIOS, this "Flash Drive" can even be made bootable.

Intel's FlashFile™ component product line is the RFA architecture of choice for mobile computers. For more information on these devices, please reference the Additional Literature section at the conclusion of this paper.

Flash Memory Cards

Mobile computers by their nature and operating environment require connectivity and code/data interchange with other computer systems. Today, two solutions exist for this purpose; neither of them is by any means an optimum approach.

The floppy disk drive is a common means today of passing data between portable and desktop computers. The 3.5" floppy is an established medium, disks are plentiful and easily available, and no added software knowledge is required beyond standard DOS commands. However, the FDD has the following disadvantages:

- The floppy disk drive, due to its rotating media motor drive, is a significant consumer of battery power in portable PCs.
- The FDD, along with the HDD, are the two most significant contributors to system height. The FDD also negatively impacts system weight.
- The FDD suffers from even longer head seek and rotation delay than the HDD. File read and write times are annoyingly long. System performance is significantly impacted
- Rotating storage media is very sensitive to the everyday "beating" and temperature extremes of the mobile environment. Storage media must be able to resist shock and vibration of large magnitude and from all axes and angles, and remain operational while non-stationary and in all possible orientations. The FDD, along with the HDD, has been shown to be inadequate to meet these specifications.
- Beyond ruggedness, magnetic media is unreliable. How many of you have ever saved a file to disk, only to be unable to access it at a later date? *Don't put your diskettes near the video display; don't put your diskettes near the power supply*...the litany of warnings is numerous! As users rely more and more on their computing platforms for data storage and manipulation (and broaden use of these computers), the value of this data exponentially increases beyond the cost of the system. Data loss is therefore unacceptable.

The other means used to interface between computers is with software such as LapLink™, which enables communication thru cable connecting serial ports. Disadvantages of serial port transfer include:

- A "free" serial port must be available on each machine (not used by a modem, mouse or printer). Given the complexity of today's computer systems, this requirement is often difficult to achieve. Hardware hookup is a complex task to many computer users.
- Serial ports must be easy to physically access for cable hookup.
- This transfer method is extremely slow, on a bit-by-bit basis (plus error-detecting bits). Performance is comparable to a modem-based file upload or download.
- Additional software commands must be learned beyond the standard DOS operations.
- In summary, serial port transfer requires knowledge of computer hardware and software beyond the level of all but computer "experts". This method is not for the "faint-of-heart"!

Does a solution exist that answers the disadvantages of the above methods? Again, the answer is flash memory; this time in card-based form!

- Flash memory cards are available in densities up to 20 Mbytes! This is plenty of room to store not only data but also full application file sets.
- The only computer hardware required to interface to a flash memory card is a low-height 68 pin connector (2.25" x 0.25" x 0.5"), an easily-integrated and compact 12V converter for flash memory write/erase, and miscellaneous interface logic. The system height and weight limitations constrained by the FDD have been eliminated.
- Solid-state storage media, by its absence of a motor and its instantaneous "on/off", is much lower power than a FDD or HDD. See the next section on The Solid State Drive for additional information.
- Flash memory cards are very high performance, with read access times of 200ns and a x16 parallel interface for high data throughput. No seek and rotation delay; no serial access!
- Solid state media is significantly more rugged than rotating media. Data stored in flash memory is intrinsically nonvolatile for 100 years!
- Microsoft's Flash File System allows flash-based storage to emulate today's FDD and HDD, with identical DOS commands. No additional software learning curve is needed.

Table 2. Flash Memory Comparison to Hard Disk Drives

Characteristic	Flash Specification	HDD Specification	Flash Advantage
Power Consumption	0.05 Watt-Hour	1 Watt-Hour	**20x**
Volume	15 Cubic cm	60 Cubic cm	**4x**
Weight	36 grams	70 grams	**2x**
Ruggedness	1000 G	10 G	**100x**
Reliability	1,000,000 Hrs MTBF	100,000 Hrs MTBF	**10x**
Time-To-First Access	200 ns	20 ms	**100,000x**
Media Transfer Rate	10 MB/sec	1.4 MB/sec	**7x**

- Card interface is simple via the integrated PCMCIA connector. Complex hardware contortions are eliminated.

Flash memory cards are writeable and updateable; capabilities lacking in ROM cards. Compared to battery-backed RAM, flash memory cards are **fully** nonvolatile (no battery to fail), higher density, and lower cost on a per-Mbyte basis. Intel's FlashFile Series 2 memory card product line offers a powerful removeable mass storage media for mobile computers. For more information on these devices, please reference the Additional Literature section at the conclusion of this paper.

The Solid State Drive

Many future pen-based computers will use flash memory cards exclusively as their mass storage systems. Where additional solid-state storage beyond the density capability of cards is needed, or where "plug-and-play" rugged storage upgrade to existing HDD designs is desired, a solid state drive is the solution of choice. The IDE- or SCSI-compatible solid state drive market is still in its infancy. Within the next year, expected product announcements from several vendors will validate the concept of flash memory for "disk drive"-like mass storage. In particular, Intel Corporation and Connor Peripherals Inc. have signed and publicly announced an agreement to jointly develop solid-state based storage products.

Table 2 details comparisons between current 1.8" hard disk drives and Intel Series II flash memory cards. Flash's dominance in all areas is clearly evident. William Schroeder, vice Chairman of Connor Peripherals, Inc., a hard disk drive manufacturer, sums it up best. "Because a mechanical drive requires a mechanical arm to move over the disk to access information, portable applications for this technology will be limited to those in which the unit is stationary when operated. For a majority of mobile computing applications that are active while moving, solid-state disks will dominate".[5]

SUMMARY

The emerging mobile computer market is the high growth segment in this industry, and offers computer manufacturers tremendous opportunities for differentiation. Success in this market requires analysis of the specific applications and their requirements, followed by selection of enabling technologies to answer these needs. ETOX flash memory is such a technology. Only ETOX flash memory satisfies all portable computing essentials: ruggedness, high reliability, light weight, compact size, low power and high performance. Whether in component, card or subsystem form, flash memory is uniquely positioned to help lead mobile computing into the future.

ADDITIONAL LITERATURE

For additional information on the Intel flash memory products mentioned in this article, please reference the following documents, available thru your local Intel sales representative.

BootBlock Components	Order Number
28F001BX Datasheet	290406
28F200BX/28F002BX Datasheet	290448
28F400BX/28F004BX Datasheet	290451
AP-341 "Designing an Updateable BIOS Using Flash Memory"	292077
AP-343 "Extended Flash BIOS Design for Portable Computers"	292098
ER-26 "The Intel 28F001BX-T and 28F001BX-B Flash Memories"	294010
ER-29 "The Intel 2/4 Mbit BootBlock Flash Memory Family"	294013
TP-355 "Flash: The Optimum BIOS Storage Device"	297003

FlashFile Components	Order Number
28F008SA Datasheet	290429
AP-359 "28F008SA Hardware Interfacing"	292094
AP-360 "28F008SA Software Drivers"	292095
AP-364 "28F008SA Automation and Algorithms"	292099
ER-27 "The Intel 28F008SA Flash Memory"	294011

FlashFile Series 2 Cards	Order Number
Series 2 Flash Memory Card Datasheet	290434
AP-361 "Implementing the Integrated Registers of the Series 2 Flash Memory Card"	292096

General Flash Information	Order Number
AP-357 "Power Supply Solutions for Flash Memory"	292092
ER-20 "ETOX™ II Flash Memory Technology"	294005
ER-28 "ETOX™ III Flash Memory Technology"	294012

[1]"Pen Computing Disappoints Judges," PC Week, July 20, 1992, p. 80.
[2]"Pen Computing Disappoints Judges," p. 80.
[3]"Pen Computing Disappoints Judges," p. 80.
[4]"Pen Computing Disappoints Judges," p. 80.
[5]"Flashy Mass Storage Challenges Magnetic in Laptop Computers," Electronic Buyers' News, September 1, 1992

Microsoft, MS-DOS and Windows are trademarks of Microsoft Corporation. OS/2 is a trademark of International Business Machines. LapLink is a trademark of Traveling Software, Inc. ETOX and FlashFile are trademarks of Intel Corporation.

BIBLIOGRAPHY

Brian Dipert is an applications engineer in the Memory Components Division at Intel Corporation in Folsom, California. He holds a BSEE degree from Purdue University in West Lafayette, Indiana.

Flash Memory For Top Speeds In Mobile Computing Applications

by **Bruce Bonner**

The access time advantage of a semiconductor mass storage technology like flash is key in giving performance benefits over a 1.8-in. rotating magnetic disk drive. Users can expect noticeable performance increases when using flash instead of hard disk drives in most cases inasmuch as access time is such a large component of normal DOS cluster read and write operations. Joined with its low power consumption, small size, light weight, silence and ruggedness, flash clearly advances miniature mass storage technology for mobile computing, and the state-of-the-art in mobile computing in general.

The time for a typical 1.8-in. hard disk drive to get the first byte of data is about 25.6 msec, whereas for the flash memory card it is 0.2 msec, more than a 100:1 ratio (Fig 1). This ratio is the reason for the success of disk caching programs such as SMARTDrive in Microsoft Windows 3.X. On a repetitive basis data is obtained from system DRAM, instead of mechanical disk, giving significant gains in apparent disk performance.

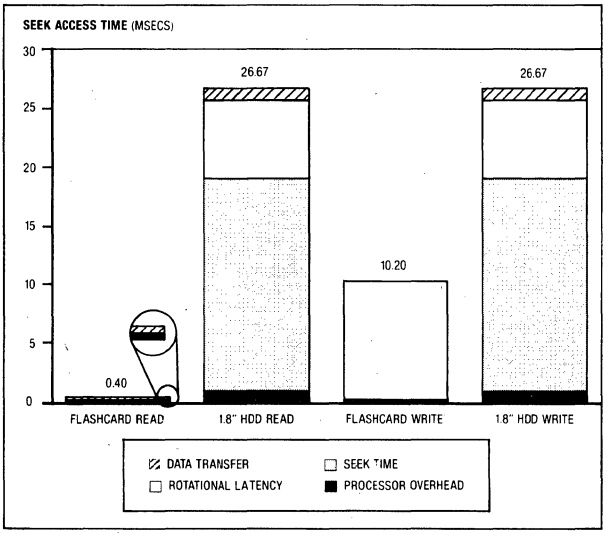


Fig 1 The seek/access time greatly affects the time required to read and write one DOS 5.0 disk cluster.

Power-Off Access Time

Disk drives consume the most energy when moving a mechanical part, such as the rotating disk(s) or heads. To conserve battery life in mobile computers the drive spins down during idle times. Typically the drive is automatically put into this 'sleep' mode after a minute or two of idle time by system BIOS power management software. (The exact idle duration time is configurable by the user.) The drive stays asleep until the user needs to read or write data to the disk, at which time it is 'spun up.' Typical hard disk drives take at least a second to do this. During idle, the hard disk drive does no productive tasks, so power expended is totally wasted energy.

The action that wakes up the drive is a system disk read or write, which is the result of the user saving or retrieving data to the disk. The user has to wait for the drive to spin up and stabilize before continuing, and makes a tradeoff between long battery life (the disk not spinning all the time) and fast access time (the disk spinning all the time).

Flash, by comparison, is 'instant on.' It takes a flash memory card 1 μsec to come out of sleep mode. Flash need only be turned on during read or write operations, saving an enormous amount of power, without asking the user to make any compromises.

Media Data Transfer Rate

Hard disk drives read and write to the rotating magnetic media at the same rate. This is determined by the rotation speed of the disk, which translates into a linear velocity under the read/write head, and the write clock frequency of the flux reversals produced by the head.

Reprinted by permission of the Publisher from the June 1992 issue of COMPUTER TECHNOLOGY REVIEW®

Order Number: 295083-001

Data Access/Transfer Rates In A Flash

This corresponds to the bits per inch (bpi) of the disk drive. Modern miniature disk drives usually have variable recording frequencies (higher for outer cylinders) to obtain a more constant bpi and higher capacities. For instance, one 1.8-in. drive has a media data transfer rate of 9.9 Mbps (1.28 Mbytes/sec) at the inner most cylinder, and 21.4 Mbps (2.675 Mbytes/sec) at the outer most cylinder. Further, magnetic disk drives directly overwrite previous data, combining erase and write functions.

Flash, being a nonvolatile semiconductor technology, uses another approach. Read, write (program) and erase are separate functions. Writing requires that the block of flash memory to be written to previously be erased to an all 1's state. One 8-Mbit flash device, for instance, has a block size of 64 Kbytes; the typical erase time of a block in the flash memory is 1.6 secs. Some would suggest that this makes flash fall short in write performance.

But erase time is not a problem, because in practice erasing is carried out as a background task so that there are always erased flash memory locations ready to accept data. In a flash memory card, multiple chips can be simultaneously executing independent erase operations, further reducing the impact of block erase time. The erase rate for a 20-Mbyte card could be as high as 800 Kbytes/sec, assuming all flash chips are erasing blocks, giving plenty of sustained bandwidth for writes.

When it comes to writing, the typical flash card media data transfer rate is 200 Kbytes/sec, which results from a 10 μsec write time for a 16-bit word. This is a sustained transfer rate, assuming that a new 200-nsec access occurs for every word. The hard disk drive's sustained data transfer rate of about 1.9 Mbytes/sec is dependent on the head staying on one track, which does not realistically happen for a file that has been modified over time; it becomes fragmented, and occupies clusters scattered over the surfaces of the disks. Not having 1:1 interleave, which is common with AT-class computers, will also drastically reduce sustained data transfer rate.

Reading data from flash is a very high speed operation, roughly the same speed as DRAM at the chip level. One flash memory card has an access time of 200-nsec, equating to a 10 Mbyte/sec 16-bit word transfer rate. Again, the disk will have about a 1.9 Mbyte/sec transfer rate dependent on having data contiguously located, which is not an limitation with flash.

Finally, combining transfer rate with access times gives the amount of time it takes to read and write data with the two technologies. On reads flash is always much faster than disk. On writes flash is faster if the amount of data transferred is less than about 5 Kbytes. This shows the assumed amount of data to be moved is a key issue. Looking to normal usage for guidance, let's assume the block size that Microsoft DOS 5.0 uses in disk accesses, which is 2 Kbytes (4×512 byte sectors). DOS in most cases treats multiple cluster accesses (such as in a large file) as separate events.

Real users both read and write data to a mass storage device. Intel's research indicates a typical user reads four times more data than is written, owing to the fact that program loading and execution is mostly a read only procedure, while saving data files are mostly write-only operations. System level results based on this user model point to an overall 11:1 benefit of using flash (Fig 2). Because every user is different, the performance benefit of flash will vary, but it will always be large since it is non-mechanical. ■

Bruce Bonner *is the SSD product manager at FlashCard Systems Group, Intel Corp. (Folsom, CA).*

intel®

AR-717

ARTICLE
REPRINT

THE MANY FACETS OF FLASH MEMORY

MARKUS A. LEVY
SENIOR TECHNICAL APPLICATIONS ENGINEER

July 1992

Order Number: 295084-001

THE MANY FACETS OF FLASH MEMORY

Markus A. Levy
Intel Corporation
1900 Prairie City Road
Folsom, CA 95630

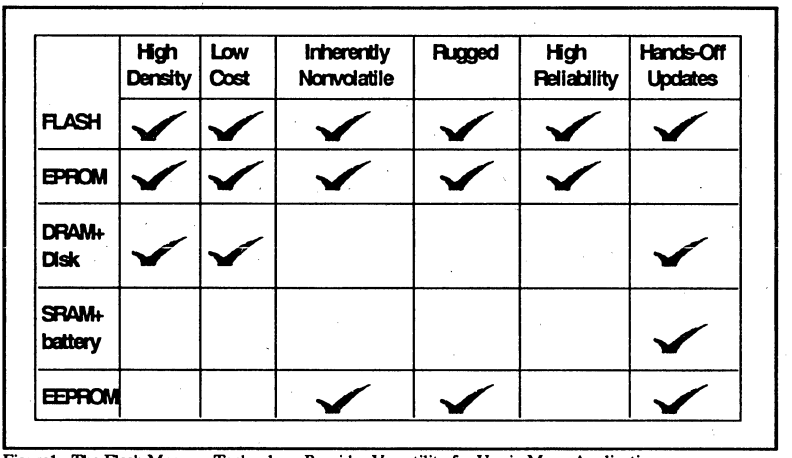
ABSTRACT

The characteristics of Flash Memory allow it to permeate into many types of applications, from code storage to disk drives. In this paper, we will discuss this memory evolution focusing in particular on using the new Series 2 Flash Memory Card as a mechanical disk drive replacement.

A MEMORY EVOLUTION

Think about all the uses for memory in a computer system. Memory for code and BIOS storage. Memory for code execution and data manipulation. Mass storage memory for user's applications and data files. Traditionally, each of these applications has been associated with one or more memory technologies. EPROMs typically handle code and BIOS storage. Disk drives store the applications and user data files that get downloaded to system DRAM during execution. In demanding environments, battery-backed SRAMs have taken the place of disk drives. Or for increased flexibility and nonvolatility in low densities, EEPROMs may replace anything from EPROMs to SRAM. When flash memory appeared in the industry a few years ago, it demonstrated the technical ability to displace each of these memory types to a varying degree (Figure 1). With properties that include nonvolatility, high density, in-circuit write and erase capability, random access, high reliability and low power consumption, flash memory's degree of utilization ties directly to cost and design considerations.

Reprinted with permission from IC Card Expo Conference Proceedings 1992. For more information about IC Card Systems and Design, please call 303-220-0600.



	High Density	Low Cost	Inherently Nonvolatile	Rugged	High Reliability	Hands-Off Updates
FLASH	✓	✓	✓	✓	✓	✓
EPROM	✓	✓	✓	✓	✓	
DRAM+ Disk	✓	✓				✓
SRAM+ battery						✓
EEPROM			✓	✓		✓

Figure1. The Flash Memory Technology Provides Versatility for Use in Many Applications.

Intel holds approximately 85% of the flash market with its ETOX (EPROM tunnel oxide) Flash Memory devices. The ETOX flash memory cell provides a very scalable lithographic process which has resulted in a very rapid increase in component density and decrease in cost. From the start, flash memory prices allowed it to replace EEPROMs and SRAMs in many applications requiring code and data updates. Despite a price differential, it quickly became obvious that flash could replace EPROMs for improving customer service by fixing software bugs and upgrading systems with easy field updates.

Meanwhile, back at the ranch, the computer industry was undergoing an evolution of its own. Smaller size, lower power, higher performance and mobility have become the new driving factors. A new breed of machines, from notebook PCs to Federal Express delivery trackers, placed new demands on system designers. Space and power constraints and the need for performance and reliability improvements are factors that make disk drives a focal point for system improvement. Although disk

drive manufacturers strive towards meeting the new requirements, many application goals can only be achieved by using solid-state devices. With the recent introduction of the Intel 28F008SA, one megabyte flash memory device, pricing came on par with DRAMs. This prompted OEMs to use flash for disk drive alternatives and even as part of system memory, especially in small form-factor portable machines.

INTRODUCING FLASH MEMORY SOLID-STATE STORAGE

In any given system, mass storage memory, in general, exists in two basic flavors: fixed and removable. Naturally, for mechanical disks, this means hard disks and floppy disks, respectively. For solid-state storage, this implies chips on the system motherboard and removable memory cards, respectively. For flash memory drives specifically, the system motherboard houses the 28F008SA devices connected directly to the CPU bus (referred to as a Resident Flash Array or RFA). This approach provides the highest performance because the AT bus does not limit access speed, as it would with removable memory cards. In this format, the flash memory array can also be used for ROM DOS and ROM WINDOWS. The true flash value becomes apparent when new software revisions arrive allowing the user to perform a simple update procedure.

In support of removable memory technologies, the Personal Computer Memory Card Association (PCMCIA) standardized the electrical and mechanical interface for memory and I/O cards. This made removable mass storage memory practical by establishing a non-proprietary interface and facilitating system-to-system transfer of data and applications via memory cards. In accordance with the PCMCIA specification, Intel developed a 20 Megabyte card, called the Series 2 Flash Memory Card. The card contains up to 20 28F008SA devices arranged in pairs to provide a 16-bit data path. The functionality of this flash memory card (and the 28F008SA), in conjunction with new software drivers,

allow it to perform high-performance disk emulation.

THE TECHNICAL SIDE OF FLASH MEMORY SOLID-STATE STORAGE

To understand the operation of a flash memory solid-state disk, we begin with a discussion of the technology. All the bytes in a flash memory device erase simultaneously, (hence the name 'FLASH') and after writing to all bytes, the device must be erased before it can be rewritten. In a typical system with a mechanical disk drive, files (or data) are constantly being written and deleted. When handling these modifications, a mechanical disk drive continuously modifies the Directory and FAT structures. Furthermore, files can be written simply by overwriting deallocated space. Unlike a mechanical disk, which can rewrite data in small, 512 byte sectors, a flash memory drive must employ unique software designs to effectively allow it to appear functionally similar.

Microsoft's Flash File System (MS-Flash) exemplifies this type of software. To overcome the Directory and FAT structure rewrite issue, MS-Flash stores and locates files using a linked-list data structure (Figure 2). With this technique, file information (name, date, time, attributes, etc) attaches directly to the file itself, rather than having a dedicated directory space. Furthermore, files only get written to clean, unused flash memory in a stack-like manner. Deleted files remain intact, to avoid continuously erasing and rewriting the flash memory card. Eventually the entire flash memory array gets used up. The real trick lies in removing the deleted files (referred to as 'taking out the garbage'). With deleted files mixed in with valid files, the valid data must be relocated (to isolate the two) as a background task, imperceptible to the user, to deliver disk-like functionality.

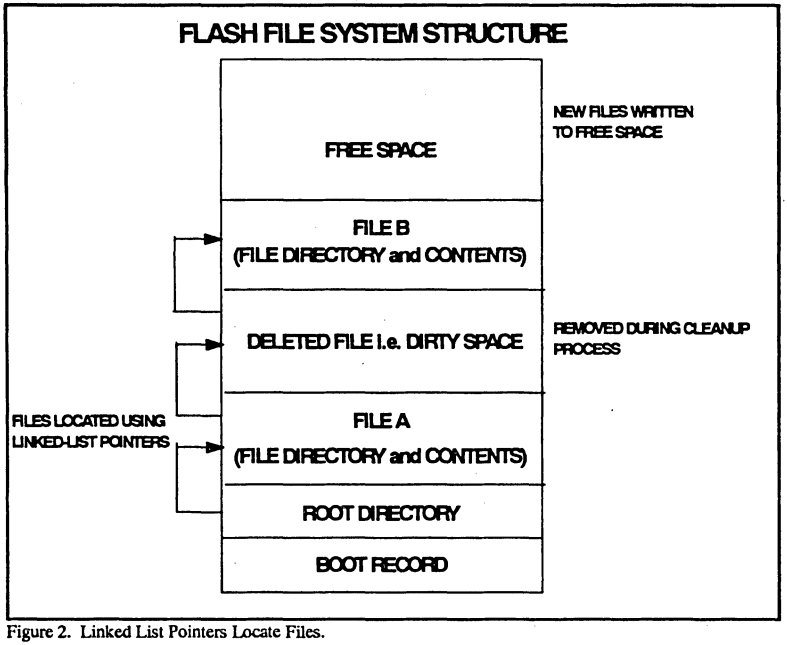


Figure 2. Linked List Pointers Locate Files.

As a first step in this garbage removal process, the software writes the valid file data into a previously reserved, spare flash device pair (referred to as a copy operation). But to make this operation practical from a memory utilization and performance standpoint, a card must contain many device pairs. The Series 2 Card features become important here. Let's start with the 28F008SA, the flash memory technology within the card. This device consists of 16 separately-erasable blocks (64Kbytes each), which means the entire device does not have to be erased before it can be rewritten. As Figure 3 displays, an erasable block actually consists of 64KWords, in device-pair format. Nevertheless, a 20 Megabyte Series 2 Card contains 160 of these erasable blocks, and reserving 2 or more of these blocks for the cleanup process has negligible impact (1% of the card's blocks).

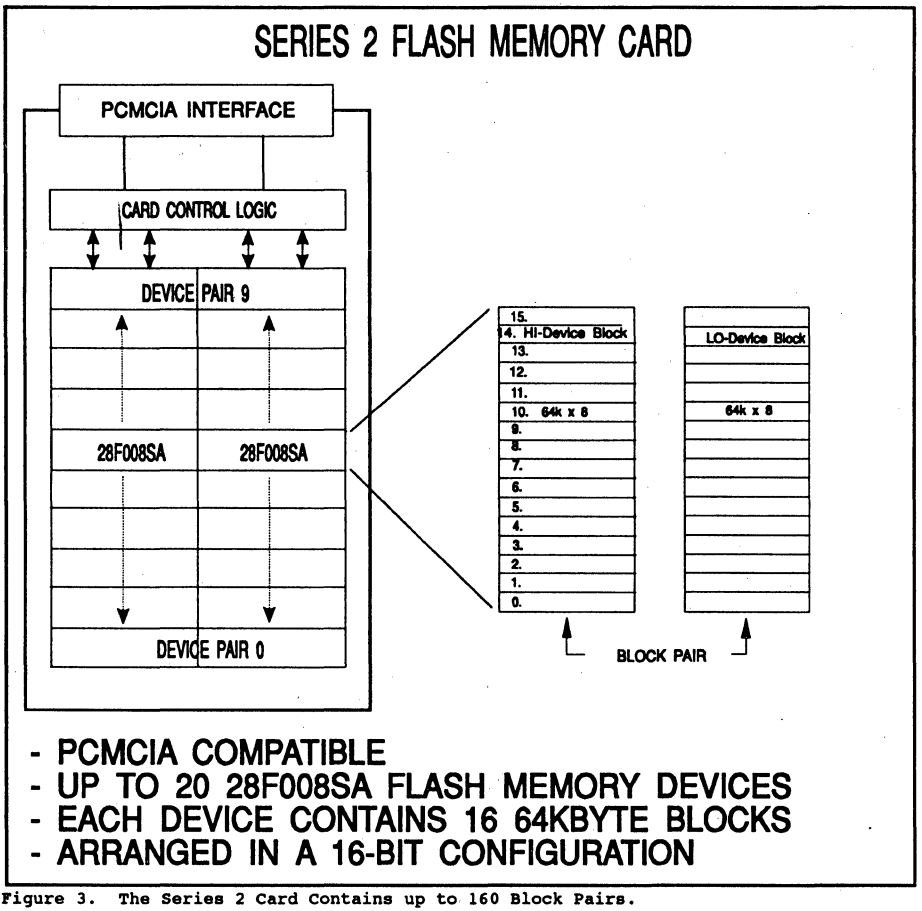


Figure 3. The Series 2 Card Contains up to 160 Block Pairs.

To facilitate the cleanup process, the 28F008SA uses built-in circuitry to automate the write operations (and block-erases), everything from timing loops to data verification. This functionality provides a performance increase and a reduced need for host system involvement. As a matter of fact, once the

28F008SA receives the appropriate command (and data), the host CPU is free to perform alternate tasks until the flash memory device signals operation complete with its READY/BUSY pin. The READY/BUSY pins from the individual 28F008SAs are wire-OR'd together within the Series 2 Card to form the PCMCIA-defined RDY/BSY output. This output can be connected to a system-level interrupt to allow asynchronous notification of a completed operation.

The copy function, discussed above, actually consists of a varying number of data writes. Typically, a single, data-write operation requires only 7·sec. Obviously, we wouldn't want an interrupt to occur after every write because the interrupt latency time would be considerably longer than the write operation itself. The Series 2 Card has a mechanism in a special mask register that selectively filters individual component's READY/BUSY pins from the card's interface. So if an interrupt is filtered, how does the system know when a data-write operation completes? Each 28F008SA has an internal status register, with a READY/BUSY bit, that can be polled periodically (Figure 4). Additionally, this register reports whether the operation was successful.

28F008SA STATUS REGISTER BIT DEFINITION

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
WSM STATUS	ERASE SUSPEND STATUS	ERASE STATUS	WRITE STATUS	VPP STATUS	R E S	E R V	E D

Figure 4. The Status Register Internal to the 28F008SA.

During the second step of the cleanup process, which creates a new spare block from the dirty, old block, software gives the dirty block the erase command (Figure 5). During this operation,

the system would not want the interrupt blocked so software ensures that the mask was not enabled for the block erasing. A block-erase operation typically takes 1 second, but once initiated, the host can go off until it receives the interrupt generated by the READY/BUSY signal making this process imperceptible to the user.

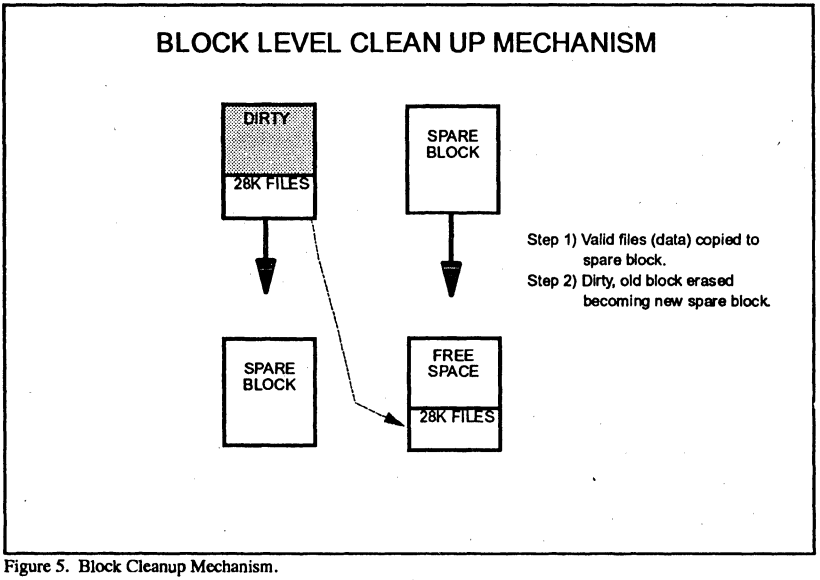


Figure 5. Block Cleanup Mechanism.

THE CONTINUING EVOLUTION

The dirty-file cleanup discussed above, is the fundamental process a flash memory drive must perform to deliver disk-like functionality. In addition, the Series 2 Card provides many disk-desired features, ranging from very low power modes to various write protection mechanisms.

Continuously improving hardware and software compression techniques will soon allow flash memory drives to approach hard disk densities and prices. Along with the technology

improvements of the 28F008SA, comes faster read access and direct code execute capability - this eliminates the disk-to-DRAM download and implies that a Series 2 Card can partially replace system memory.

In the future, OEMs will break paradigms and begin designing computer systems without the concern for maintaining compatibility with historical, traditional technologies. Just like the evolution from card readers to tape to mechanical disk drives, flash memory and new software drivers, such as Microsoft's Flash File System, will continue to support evolution, and systems will continue to be optimized for the newer and better technology provided by flash memory.

STANDARDIZING ON A FLASH FILE SYSTEM

Markus Levy
Intel Corporation
1900 Prairie City Road
Folsom, California 95630

Wink Saville
Saville Associates
4425 Esta Lane
Soquel, CA 95073
408-479-7199

****

THE NEED FOR A STANDARD FLASH FILE SYSTEM

Why do we bother with standards? How do standards get established? More specifically, how should we choose a file system standard for flash memory that will enable the visions of PCMCIA for interchanging PC cards between sockets of different systems? Ultimately, this is the benefit of having standardized physical and electrical interfaces. But these interfaces are of minimal value when the data format of the information within the flash memory card has not been standardized (Figure 1). This issue, among others, represents the subject of this paper in which we will try to provide you with an appreciation for establishing a standard media manager and data structures for removable flash memory cards.

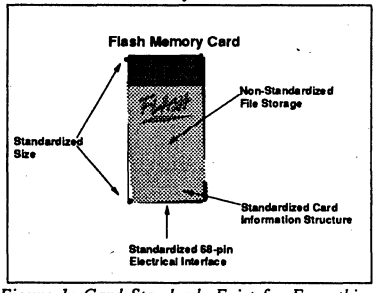


Figure 1: Card Standards Exist for Everything but the File Storage.

As our world becomes more and more electronic information intensive, the need to share information across many environments becomes a necessity. The communication industry is rapidly promoting a new class of computer architecture using a myriad of processors and operating systems. Two main requirements must be fulfilled in order to provide a common link and share information:

- First, and foremost, the information, or file contents, must be in a sharable form. The scope of this situation certainly goes beyond the boundaries of one standard, but various attempts have been made. Examples of these standards include Postscript, TIFF, SGML, RTF (Rich Text Format), Quicktime, and AVI (Audio Video Interleave).

- Standardized file structures must also be used for removable media to support the progression of the standardization of file contents. An example, familiar to most, is the directory and FAT structures of DOS. A container can be used to give a good analogy to this situation - The information is the contents of the container and the file structures are the container itself. We need a standard container so that all equipment, or operating systems, can access the information.

WHY USE A FLASH FILE SYSTEM

The flash memory technology has witnessed an incredibly fast acceptance in the industry as an excellent media for solid-state information storage and sharing. It can replace RAM-based drives because of its lower cost and higher densities. It can replace ROM-based drives because of its rewritability and flexibility. It can even replace mechanical drives in systems where performance, power, reliability, and size are a concern.

Regardless of the memory technology, software must be used to link the media to the operating system. The device drivers available to build a RAM drive allows it to emulate the behavior of mechanical drives, even down to the level of partitioning the media into sectors. The

Reprinted with permission from IC Card Expo Conference Proceedings 1993. For more information about IC Card Systems and Design, please call 303-220-0600.

Order Number: 295085-001

functionality of flash memory differs significantly from the memories that traditionally perform disk emulation. This leads us into a brief explanation of why flash memory needs a special file system.

From a read perspective, flash memory has the random access behavior of RAM. The bytes within a flash memory device may also be written in a random fashion. But, once a bit changes from 1 to 0 (referred, to as the programmed state), the entire block must be erased to convert a 0 back to a 1. You may immediately notice that this does not resemble the mechanical drive model (Figure 2). In a mechanical drive, data may only be read or written in units of a sector, nominally 512 bytes. This is significantly different from flash memory, which may be read and written to on a byte level and must be erased on a block level to be rewritten. This means that the standard sector-based file system cannot be directly utilized on flash memory.

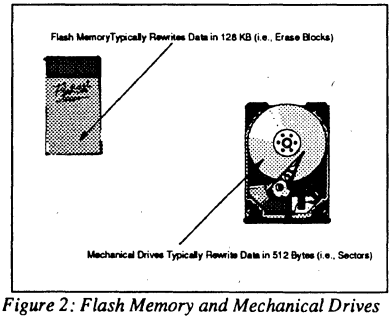


Figure 2: Flash Memory and Mechanical Drives Have Different Structures.

CHOOSING A FLASH FILE SYSTEM

Even though flash memory has proven its technical merits, computer designers still use disk-drive compatible file systems on a media that only remotely resembles a disk drive. These can be referred to as disk-drive emulators. On the other hand, Microsoft's Flash File System, specifically designed to utilize the capabilities of flash memory, has also found a place in its share of computers. It may not seem like a problem having a variety of flash file systems available, and in some cases it's not.
However, when the flash memory resides in removable IC cards that can be transferred between different computer systems, this file-system smorgasbord has the potential of creating a disconcerting amount of incompatibilities.

The selection of a flash file system should be based on an educated decision. Three fundamental approaches can be taken for choosing a flash file system:

1. Ignore any concerns for incompatibility and 'let the market' decide. The choice for a file system can be made strictly on marketing persuasions and measured performance differences. Actually, this approach is the most popular choice but it does not represent the best interest of flash users and manufacturers. Furthermore, this approach will only stall the acceptance of flash memory.

2. Use a disk-drive emulator to mimic the sector scheme of the mechanical disk. This approach has the advantage of utilizing the more traditional file structures (e.g., sectors and file allocation tables). In the DOS world, it also provides compatibility with some of the disk drive service routines (e.g., INT 13H), although the majority of applications only use Interrupt 21H. Several problems exist with disk-drive emulators:

   a. They use file structures specific to the operating system that it runs under. This means that the implementation will not take advantage of the unique characteristics of flash memory.

   b. Multiple versions of disk-drive emulators on the market indicates the lack of compatibility, even within computer systems running the same operating system. Each vendor has created unique data structures to manipulate the flash memory. In other words, a flash memory card formatted for 'Disk Emulator A' may not be compatible in the same system running 'Disk Emulator B'.

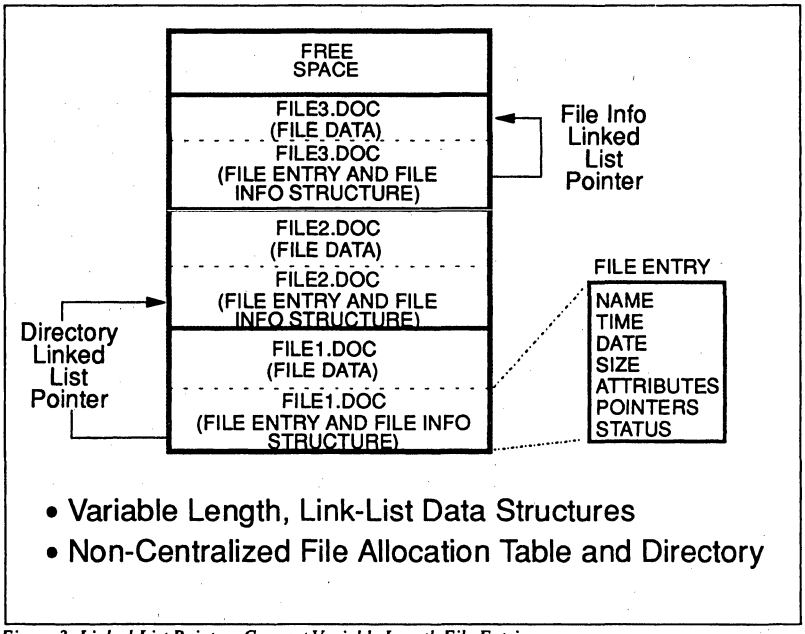


Figure 3: Linked-List Pointers Connect Variable Length File Entries.

c. Operating specific file structures in general are not designed to support attributes, which other operating systems may require. By designing an extensible file system, instead of using a lowest common denominator file system such as DOS, we can use one set of file structures for most operating systems.

d. They do not support background cleanup or wear leveling on the flash memory because they only perform tasks that the operating system requests them to do. Background cleanup and wear leveling require a file system that has 'insight' into managing the media. This can best be accomplished through a redirected file system.

3. Use an installable and extensible file system that optimizes the capabilities of flash memory without forcing it to behave like a mechanical disk. This approach is used in the design and implementation of Microsoft's Flash File System. In DOS, this is implemented as a redirector, similar to a network drive or CD ROM. Information stored with this file system is located using variable length, link-list data structures stored in non-centralized directory entries. This eliminates set sector sizes which indirectly helps to minimize fragmentation, and maximizes the efficient use of flash memory (Figure 3). The primary problem with this approach is that not all systems have documented their installable file system's interface. This will temporarily prevent developers from writing implementations that these systems require.

The First Step Towards Standardization

Microsoft has made its data structures publicly available to allow the creation of 'custom' implementations. This implies that any flash memory card formatted with these data structures can be used in any computer system using a Microsoft Flash File System compatible implementation, regardless of that system's processor or operating system. This has already

been demonstrated by Saville Associates who developed an implementation that interfaces to GO's Penpoint operating system. Although the effort to develop this implementation was non-trivial, this initial work will facilitate the development of other implementations for other environments. Ultimately, implementations of the Microsoft Flash File System will become available on most platforms, therefore, becoming the defacto standard.

As the need for inter-system communication increases, it will become more and more important to have a standardized flash file system in place. Regardless of any of the opinions stated above, it only matters that the industry, as a whole, make a choice and allow that choice to become the standard.

intel®

AR-723

ARTICLE
REPRINT

Interfacing BootBlock Flash Memories to the MCS® 96 Family

BRIAN DIPECT
APPLICATIONS ENGINEER

Reprinted From Embedded Applications Journal
Q1, 1993

Order Number: 295090-001

Interfacing BootBlock Flash Memories to the MCS® 96 Family

by Brian Dipert
Flash Memory Applications Engineer
Intel Corporation

Overview

This article reviews Intel BootBlock flash memory interface to MCS® 96 embedded processors. Three specific examples will illustrate various techniques and options available to the system designer. The article will also discuss general recommendations for all MCS 96/flash designs.

Intel's MCS 96 products, in conjunction with the BootBlock flash memory family, provide a powerful processor/memory combination for today's embedded control designs. MCS 96 devices are high-performance 16-bit microcontrollers with integrated memory and peripherals, including ports, timers, pulse width modulators and A/D converters. They easily handle high speed calculations and fast input/output operations.

Flash memory brings easy updateability to designs that in the past might have used ROM or EPROM for system code storage. Unlike battery-backed SRAM, flash memory is fully nonvolatile. Flash memory has demonstrated orders of magnitude better reliability, and much higher density, than EEPROM. Flash memory, with its much simpler cell structure, is also cheaper on a cost-per-bit basis than either SRAM or EEPROM.

BootBlock flash memory products have been specifically defined for the requirements of embedded control code storage. Features of these devices include:

- Blocked architecture (including a hardware-lockable "boot" block) integrates functions of multiple memories in one device
  - Boot block to replace boot ROM/EPROM
  - Main block to upgrade bulk-erase flash functionality
  - Two parameter blocks replace EEPROM, battery-backed SRAM
- x16 interfaces (on some devices)
  - High bandwidth read/write
- Multiple package proliferations
  - TSOP for space-constrained designs
- Fast read access time
  - High performance
- Automated write and erase
  - Simple update algorithms
- 3.3V-read operation (on some devices), and
- Multiple low power operation modes.
  Ideal for battery-operated systems

80C198 Design Example

This design example interfaces the 28F001BX-B120 to the 80C198-16. The 80C198-16 is a 16 MHz member of the MCS 96 family, with an 8-bit external data bus and an integrated A/D converter. The 28F001BX-B120 is a 120ns tACC version of the 128 KByte x8 BootBlock flash memory family, with the boot block located at the bottom of the memory map (see Figure 1). A system diagram is shown in figure 2, with a corresponding system memory map in figure 3. This example highlights the following design techniques:

- Programmable logic utilization for generating higher-order flash memory addresses
- Flash memory address inversion to match the MCS 96 boot location
- Integrating external SRAM in the design

Higher-Order Address Generation

Most members of the MCS 96 family are address pin-constrained to a 64 KByte memory map. To access the 128 KByte 28F001BX, plus any other external memory/peripheral devices, upper addresses are generated using the programmable logic device U4 as shown in Figures 2 and 3. Flash memory addresses A16-15 powerup and reset to "0"s, and are changed by writing the desired value to an internal two-bit "register" within the EPLD, using data bits D1-0 . An alternate sources of upper address bits is the High Speed Outputs (HSO3-0) of the 80C198, if not used elsewhere in the design. As shown in Figure 3, 80C198 register, port and interrupt vector location are common to all "pages".

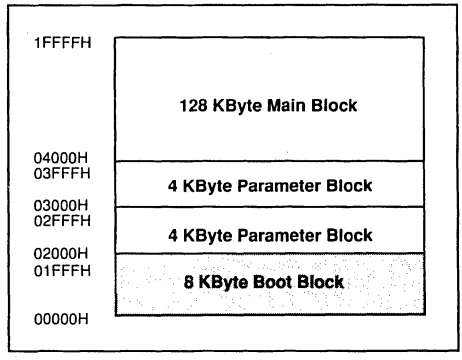


Figure 1. 28F001BX-B Device Memory Map

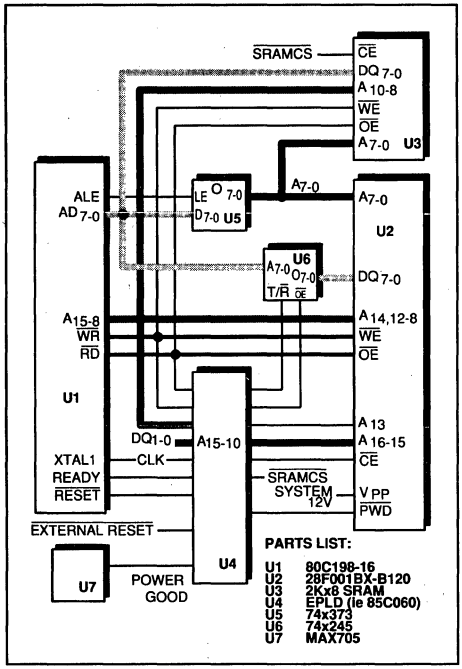


Figure 2. 80C198/28F001Bx-B System Interface

Flash Memory Address Inversion

The 28F001BX boot block is intended to store core unchanging system initialization code, port pin "page control" software and the flash memory program/erase algorithms. On the 28F001BX-B, it is located at device addresses 0000h-1FFFh (see Figure 1). Inverting address A13 with component U4 alters the 28F001BX-B memory map as it appears to the 80C198, as shown in Figure 4. This "moves" the boot block to addresses 2000h-3FFFh, compatible with the 80C198 boot/reset address of 2080h. Software developers take note: this approach also "moves" 8 KByte main block segments.

External SRAM

Component U3 is a 2Kx8 SRAM, with the following uses:

- Data storage beyond the 80C198 internal register capacity
- Code storage for execution of port pin "page control" software
- Code storage for execution of flash memory update software

For the latter two uses, the software must be executed external to the flash memory, either to change flash memory "pages" or to update flash memory contents. Therefore, the SRAM must be commonly located in every 64 Kbyte memory "page". The logic of component U4 that generates CE for the SRAM, being independent of the state of 28F001BX "addresses" A15 and A16, ensures this (see Figure 3).

The code that executes port pin "page control" and flash memory update software is stored in the 28F001BX boot block, and is copied to the external SRAM. A "JUMP" to the SRAM memory address begins execution.

Miscellaneous Details

Latch U5 latches address A7-0 for use by the 28F001BX and SRAM. Buffer U6 eliminates bus contention as the SRAM is selected and the 28F001BX is deselected, due to the fast data bus "enable" of the SRAM compared to the bus "release" delay of the flash memory.

Table 1 details the external memory timings of the 80C198-16 (along with the external logic shown) at various wait states, and the corresponding timings of the 28F001BX-B120. As can be seen, this is a 1 wait state design.

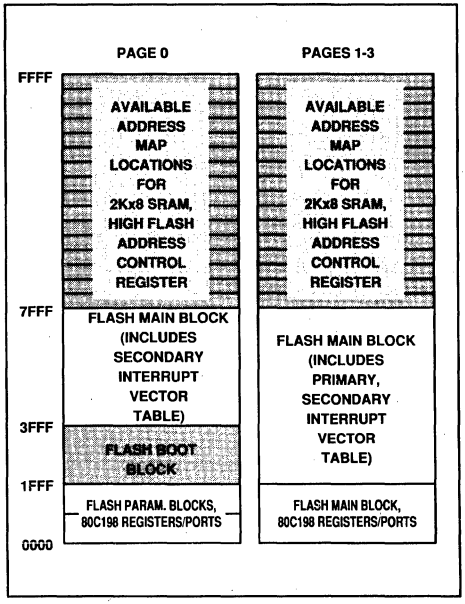


Figure 3. 80C198/28F001BX-B System Memory Map

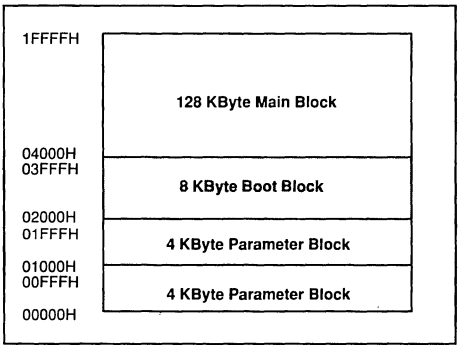


Figure 4. 28F001BX-B Memory Map (as it appears to the system after A13 address inversion)

If higher density flash memory is needed in this x8 design, the 256 KByte 28F002BX or 512 KByte 28F004BX can be used. Both of these devices offer 60ns speed bins, for high-performance no-wait-state read access.

80C196KR Family Design Overview

The 80C196KR product family consists of four devices (80C196KR, 80C196KQ, 80C196JQ and 80C196JQ) that differ in the size of SRAM and the number of integrated peripheral devices. They all communicate with external memory and logic through a 16-bit external data bus. If the 80C196KR family is used for the embedded processor, interface to flash memory is much simpler than in the earlier 80C198 example.

These microcontrollers integrate not only data SRAM but also code SRAM on-chip, in 128 or 256 byte densities. This code SRAM size is sufficient to store "page control" and flash update software. Therefore, external SRAM is only needed if internal data SRAM size is insufficient for the given application.

The 28F200BX or 28F400BX are the recommended BootBlock flash memory devices for the 80C196KR family. These flash memories both have x16 data bus interfaces, and their 60ns read speeds enable high-performance no-wait-state access. Both the 28F200BX and 28F400BX (see Figures 5 and 6) integrate 16 KByte boot blocks located at

11

Specification	80C198-16/Logic Timings (0 WS)	80C198-16/Logic Timings (1 WS)	28F001BX-B120 Timings	Unit
t_{ACC} (Address to Output Delay)	110.5	235.5	120	ns
t_{CE} (Chip Enable to Output)	110.5	235.5	120	ns
t_{OE} (Output Enable to Output)	32.5	157.5	50	ns
t_{ELWL} (Chip Enable Setup to Write Enable Going Low)	90	215	10	ns
t_{WLWH} (Write Enable Pulse Width)	47.5	172.5	50	ns
t_{AVWH} (Address Setup to Write Enable Going High)	137.5	262.5	50	ns
t_{DVWH} (Data Setup to Write Enable Going High)	32.5	157.5	50	ns
t_{WHDX} (Data Hold from Write Enable High)	10	10	10	ns
t_{WHAX} (Address Hold from Write Enable High)	32.5	32.5	10	ns
t_{WHEH} (Chip Enable Hold from Write Enable High)	32.5	32.5	10	ns

Table 1. Timing Analysis for 80C198-16/28F001BX-B120 Design (See Figure 2)

device addresses 0000h-3FFFh. This overlaps the MCS 96 2080h boot address, so no address "swapping" to relocate the boot block is required.

The 80C196KR family, like the 80C198, is address pin-constrained to a 64 KByte memory map. As in the 80C198 design example earlier, alternate means are used to "construct" upper system address bits and access the full density of the 28F200BX or 28F400BX. In this case, the port 1 outputs, if available, provide a straightforward means of implementing upper address bits. Since they power up and reset to "1"s, the port pins are inverted, as shown in Figure 7, to allow system boot from page "0" (see Figure 3).

8xC196NT Family Design Overview

The newly-introduced 8xC196NT and 8xC196NQ are upgraded versions of the 80C19KR. They retain the 80C196KR integrated code SRAM, and in addition include additional address pins. This raises the amount of directly-accessed external memory from 64 KBytes to 1 MByte, more than sufficient to handle the density of Intel's BootBlock flash memory devices. System address reconstruction using port pins is not needed in 8xC196NT family designs. Again, the recommended flash memories for these designs are the 28F200BX and 28F400BX.

General MCS 96 Flash Memory Interface Recommendations

The following tips are common design techniques for all MCS 96 microcontroller designs.

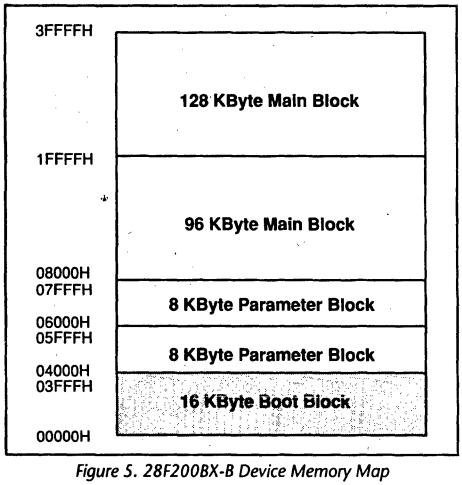


Figure 5. 28F200BX-B Device Memory Map

Redirection of Interrupt Vectors

The design example and overviews described earlier all map the common MCS 96 boot location of 2080h to the boot block of the appropriate BootBlock flash memory. The boot block is hardware-lockable via the flash memory PWD input, and code in the boot block is not normally in-system updated. The boot block also stores the lower and upper interrupt vector tables, at system addresses 2000h-2013h and 2030h-203Fh, respectively. As flash memory system code is updated, the starting address locations of interrupt service routines correspondingly change. To encompass this change, point the primarily interrupt vector table locations (in the boot block) to a secondary interrupt vector table at a fixed location in an updateable main or parameter block. As the system code is updated, this secondary vector table is also updated to reflect the new interrupt routine starting address locations.

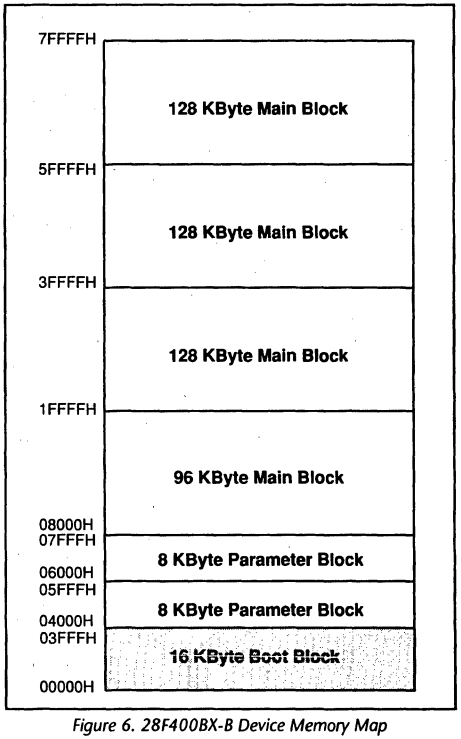


Figure 6. 28F400BX-B Device Memory Map

12V Converters

Intel flash memory devices require 12V ± 5% for byte program and block erase. Where a voltage supply with these specifications is not already available to service other components of the system, a broad range of 12V converters is available that translate existing power supply voltages (higher or lower) to meet flash memory requirements. These converters have been optimized around one or several of the following key parameters:

- Current output
- Integration
- Minimal board space
- Lowest cost

Application note AP-357 "Power Supply Solutions for Flash Memory" (order number #292092) compares a wide range of 12V solutions against the above parameters. It is available from your local Intel or distributor sales office. The following vendors are representative of those offering 12V converters.

Linear Technology Corporation

1630 McCarthy Blvd.
Milpitas, CA 95035-7487
(408) 432-1900

Maxim Integrated Products

120 San Gabriel Drive
Sunnyvale, CA 94086
(408) 737-7600

Motorola Semiconductor Inc.

616 West 24th St.
Tempe, AZ 85282
(800) 521-6274

National Semiconductor

2900 Semiconductor Dr.
P.O. Box 58090
Santa Clara, CA 95052
(408) 721-5000

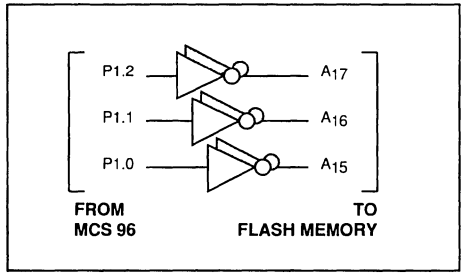


Figure 7. High-Order Flash Memory Address Generation Using MCS 96 Port Pins

PWD Control

The PWD input in BootBlock devices has several functions:

- "Unlocks" the boot block, when brought to 12V, for program/erase of this block
- Puts the device in a very low current draw "Deep Powerdown" mode when brought to VIL
- Terminates any BootBlock automation currently in progress when it transitions to VIL, and resets the flash memory to enable read of array data
- Blocks any unwanted writes to the flash memory during system powerup and powerdown, when at VIL

The design example shown in Figure 2 gates the BootBlock PWD input with a system RESET generated by any external RESET input (if present), and by the output of the MAX705 (U7). The MAX705 is a VCC monitoring circuit available from Maxim Integrated Products, and is representative of comparable offerings by several semiconductor vendors. If power consumption is a concern in the end design, the PWD input can be further gated by a MCS 96 port pin, to access Deep Powerdown mode. Since the boot block is typically programmed before the flash memory is soldered on the system board, no provision for 12V on the PWD input is shown in Figure 2. A hardware jumper is one way of accomplishing this, if required.

For more information on the products or concepts discussed in this article, please contact your local Intel or distributor sales office.

Flash Memory

Since the early 1970s, semiconductor memory devices have helped pave the way for smaller, lighter, more sophisticated computing products. The most compelling of these is flash memory—a technology in which Intel has remained the market-segment leader for over seven years.

Intel Flash offers design engineers an entirely new memory alternative. Because flash is a high density, nonvolatile read/write technology, it is both a cost-effective and reliable replacement for EPROMs, EEPROMs, DRAMs, battery-backed RAM and rotating media. Flash's inherent advantages over these technologies make it especially useful in portable systems that require low power, compact size and ruggedness while maintaining solid-state performance and full functionality.

This databook is devoted to techniques and information to help design flash memory into an application or system. Informative datasheets provide many comprehensive charts, block diagrams, operating characteristics and programming modes. Application notes provide diagrams and hardware design information. Relevant article reprints are also included.

intel

Order Number: 210830-014
Printed in USA/0195/21K/RRD/TS
Memory Products

Printed on Recycled Material

ISBN 1-55512-233-7

9 781555 122331 90000